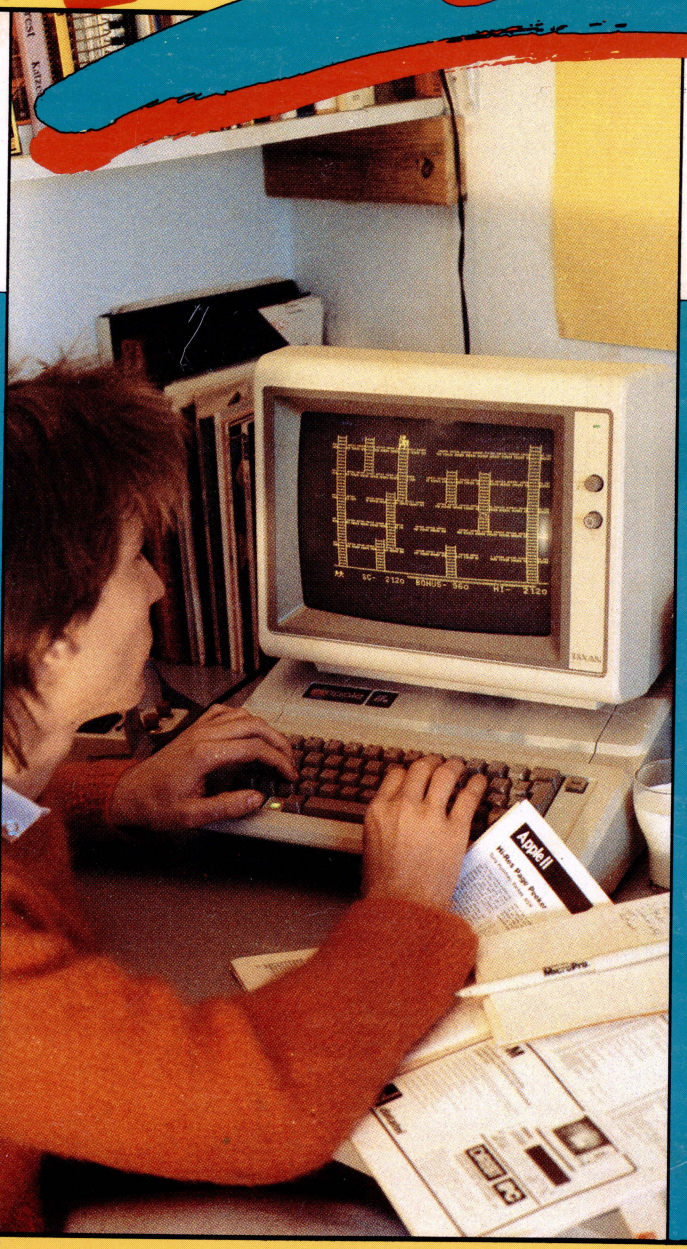
$4.50* NZ$5.50

A Your Computer Publication

Bumper Book of Programs

PROGRAMS FOR ALL AGES

- Games
- Business
- Useful Utilities
- Programming Tips
- Computer Club List



ALL POPULAR BRANDS

- Apple
- Commodore
- Dick Smith
- Hewlett Packard
- Microbee
- Sharp
- Sinclair
- Tandy
- and many more

CONTENTS

Editor
Evan McHugh

Cover Design
Ali White

Art and Production
Brian Jones

Managing Editor
Jamieson Rowe

Advertising
Richard Pakalnis (Group Advertising Manager)

David McDowell (National Advertising Manager)

Mel Godfrey (Victorian Advertising Manager)

The Your Computer Bumper Book of Programs is published by the Federal Publishing Company Pty Ltd. Printed by ESN-The Litho Centre, Waterloo 2017. Phone (02) 662-8888.

Editorial and NSW Advertising: 140 Joynton Avenue, Waterloo 2017. Phone (02) 663-9999. Postal address: PO Box 227, Waterloo 2017. Telex: FEDPUB AA74488.

Publisher: Michael Hannan. Typeset by Hughes Phototype, Spit Junction NSW. Phone (02) 960-2788. Distributed nationally by Gordon & Gotch.
ISBN No. 0 86405 081X
*Recommended and maximum price only.

INTRODUCTION	**4**
APPLE	**7**
ATARI	**61**
COMMODORE	**63**
HEWLETT-PACKARD	**85**
VIC-20	**87**
MICROBEE	**93**
APF	**121**
SORCERER	**122**
TRS-80	**125**
SHARP	**139**
VZ-200	**145**
SINCLAIR	**148**
SIGMA OKI	**153**
MISC: HITACHI PEACH CASIO OHIO	**156**
COMPUTER CLUB LIST	**159**

INTRODUCTION

HI, AND WELCOME to this bumper book of programs. *Your Computer* began publishing readers' programs in a special feature, called Pocket Programs, back in December 1982. They have been incredibly popular with our readers ever since and now appear every second month.

Happily, the supply of programs has outstripped our demand, which means we have a never-ending stock of good-quality programs to offer other readers. Recently, however, we have been embarrassed by the backlog. Programs were coming in much faster than we could print them and we were beginning to get buried under a growing pile of listings and documentation.

The solution was obvious: let's get all the programs together and publish the lot in one big collection. The contributors get to see their programs in print, you get a whole swag of programs to try out on your computer and we get to see what our desks look like.

Anyway, here they are: programs for the beginner through to the advanced; technical programs and games programs; programs from kids aged ten up to adults aged seventy. Programs in a variety of languages for all the popular machines and the more obscure ones as well. You will find lots of programs to use on your computer straightaway, and lots more that you can adapt from other computers and languages.

You will see we have divided the programs according to the machine they're written for: Apple, Commodore, Sinclair and so on. Don't restrict yourself just to looking at the brand you own. Many of the other machines' programs can easily be changed to run on different computers, and there are notes in some to suggest how you might go about doing this. In the miscellaneous section there are programs you can check out that are written more for a particular language than a particular machine.

There are also a few tricks to typing in the program that might be useful. Here are some hints I have found useful.

First, place a ruler under the line you are typing to mark your place. You don't want to start typing the wrong line midway through another, or leave a couple out. The results can be catastrophic, almost as confusing as that last sentence.

Next, check the data in data statements very carefully. When you type in normal commands and make a mistake it is usually pretty obvious. For example:

```
IB A>0 THEN ABORT
```

is a lot easier to correct than, say:

```
1000 DATA 143,233,233,087,323
```

One check you can make is to count how many numbers you should have typed in and how many you have typed in. You can also get someone to help you by reading the data to you while you type it in. If nothing else it makes for a more social occasion and makes your husband or wife, mother or father, feel wanted.

When you have typed in a program it may return an error. When you discover the line causing the error don't just check it and think "That looks OK". The best thing to do is read the line backwards, letter for letter and check each letter, number or control code against what you have typed in. That way you don't assume that everything is correct as you quickly flick your eye over the line. You plod through and verify every single character.

If you're still having trouble finding an error, another trick is to put a trace on the execution to follow the path the program takes. If your program loops uncontrollably you can use a command (TRON in many BASICs) which will show you the line

By Evan McHugh

numbers as they are being executed. Another thing you can do if you don't want to trace through the whole program is to sprinkle PRINT statements throughout or in selected locations. These can tell you all sorts of things about the execution. They can just say, "Hi, I'm at line 100 and everything is fine!", or they can tell you the value of the variable that seems to be causing the crash: "The value of C is 20."

With these few debugging tools, hopefully you should be able to work out about 99 per cent of the problems you may face. Of course, there is always that worst of bugs, the invisible, undetectable bug. These little monsters will have you tearing your hair out, glaring at your screen until three in the morning and in spite of your best efforts will never make themselves apparent. Often such bugs will cause you to despair, sell your computer and go on a skiing holiday to Europe with the proceeds.

It happens to every programmer from rank beginner to seasoned professional. For example, one of my computing lecturers was frowning at a listing a student had brought to her for some help. Another student noticed the frown and offered assistance.

"The bug must be in this line," said the lecturer, "but I've been looking at the rotten thing for two hours and there is nothing wrong with it. We've looked at everything; whatever it is must be pretty weird to cause an error." "Let's have a peep then," chirps the helpful student. "Ah yes, that comma should come before the variable, not after."

It had taken him three seconds to find the bug. It is times like that when quite talented people can get turned onto the alternative lifestyle, but please don't despair. Sometimes things can be extremely complicated with computers, but far more often they are extremely simple. The solution is to get a second opinion. Another programmer used to get his kids to check his syntax when he ran into an error. They didn't know a thing about computers, but if he explained the way syntax worked they could pick up the obvious mistakes which he had looked at for hours without noticing.

Also, you should consider joining a computing club. You will certainly meet lots of people who will gladly take a look at a listing and point out any bugs that might be causing trouble. A full listing of clubs in Australia and New Zealand is printed in this book, and we update it from time to time in *Your Computer*.

If you still can't find the bug after trying all these avenues it is time to despair, sell your computer and buy a sailboard with the proceeds.

Hopefully, having tried some of the programs in this magazine you will be inspired to write some programs of your own that you would like to submit to us for publication. Please feel free to do so. The programs we like best are ones that have some creativity about them. Say, a new way of performing an old routine, or a game that has not been put on a computer before, or a useful routine that works faster than any that are around at the moment.

If you are a rank beginner don't think there are no programs you can send in. There are plenty of other rank beginners out there who will probably find that your programs are just at the level they can understand. So, send those programs rolling in.

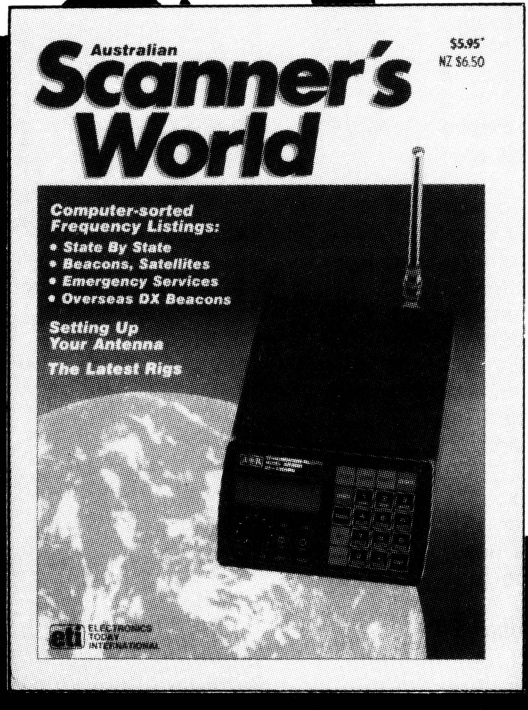
We hope you have lots of fun with the programs in this magazine. There is something for everyone. And, as my desk diary for today says, "You should try everything once, except incest and folk dancing", which I'll admit is a trifle weird, but it's not a bad approach to trying out the programs!

Enjoy and Keep On Computing! ■

GET AMONGST THE ACTION!

There's a whole world of 'action' on the bands between 30 MHz and 500 MHz. No matter whether you're interested in VHF/UHF DX, or just the local 'chatter', a scanner will put you 'in touch' with that world of action.

Australian SCANNER'S WORLD is the book that will introduce you to that other world 'beyond the shortwaves'. It contains an introduction to scanning and scanners, an article on scanner antennas — including how to build two types for yourself, along with how to erect antennas. The major part of this book is the "Listener's Guide", computer — sorted listings of services throughout Australia and New Zealand, with their frequencies listed in both frequency order and alphabetical order by service. Beacons are listed also, along with relevant overseas ones. A roundup of scanners, antennas and accessories is also included.



FIND OUT WHERE THE ACTION IS!

Australian SCANNER'S WORLD $5.95

at your newsagent

PROGRAMS FOR APPLE II

FROGGER

Frogger is a two part program, connected by the CHAIN program on the DOS 3.3 SYSTEM MASTER. The listing REM FROGGER should be saved under the name of "FROGGER", and the other under "FROGGER@". A copy of CHAIN is expected to be present on the disk when FROGGER is run.

To guide the frog:

A – up, J – left, K – right, Z – down.

M.J. Smith
Waramanga ACT

```
0  REM     <<<FROGGER>>>
1 DS = 2
5  GOSUB 8000
6  ROT= 0: SCALE= 1: HCOLOR= 3
7 J = J + 1
8 FB = 130:FC = 140
10 S1 = 10:S2 = 20:S3 = 8:S4 = 12
20 FB = 130:FC = 140
30 D$ =  CHR$ (4)
40  GOSUB 20000
50  POKE 232,0: POKE 233,64
60 AB = 90:AC = 200:AD = 60:AE = 180:AF = 210:AG = 50:AH = 210:AI = 40
70 S1 = S1 + J:S2 = S2 + J:S3 = S3 + J:S4 = S4 + J
75  GOTO 350
80  POKE  - 16368,0
85 YF = FC:XF = FB
90  IF X = 218 THEN FD = FD - 1: IF FD < 0 THEN FD = 0
95  IF X = 218 THEN 140
100  IF X = 193 THEN FD = FD + 1: IF FD = 5 THEN 10000
105  IF X = 193 THEN 140
110  IF X = 203 THEN FB = FB + 20: IF FB > 250 THEN FB = 250
115  IF X = 203 THEN 140
120  IF X = 202 THEN FB = FB - 20: IF FB < 30 THEN FB = 30
125  IF X = 202 THEN 140
130  RETURN
140  IF FD = 0 THEN FC = 140
150  IF FD = 1 THEN FC = 112
160  IF FD = 2 THEN FC = 85
170  IF FD = 3 THEN FC = 59
180  IF FD = 4 THEN FC = 31
190  ROT= 0: SCALE= 1
192  HCOLOR= 4
195  DRAW 1 AT XF,YF
200  HCOLOR= 1
210  DRAW 1 AT FB,FC
215 SC = SC + 10
220  POKE  - 16368,0
230  IF FD = 0 THEN  RETURN
240  IF FD = 1 THEN F1 = AB:F2 = AC
250  IF FD = 2 THEN F1 = AD:F2 = AD
260  IF FD = 3 THEN F1 = AF:F2 = AG
270  IF FD = 4 THEN F1 = AH:F2 = AI
280  IF FB > F1 - 27 AND FB < F1 + 27 THEN 330
290  IF FB > F2 - 27 AND FB < F2 + 27 THEN 330
300  RETURN
330  GOSUB 2000
335 FB = 130:FC = 140
340  GOTO 350
350 FD = 0
360  HGR
370  ROT= 0: SCALE= 1
380  HCOLOR= 4: DRAW 2 AT AB,110: DRAW 2 AT AC,110
390 AB = AB + S1:AC = AC + S1: IF AB > 279 THEN AB = 0
400  IF AC > 279 THEN AC = 0
410  HCOLOR= 7: DRAW 2 AT AB,110: DRAW 2 AT AC,110
412  IF FD = 1 THEN  GOSUB 230
415 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
420  HCOLOR= 4: ROT= 32: DRAW 2 AT AD,96: ROT= 0
```

```
430 AD = AD - S2: IF AD < 0 THEN AD = 279
440  HCOLOR= 7: ROT= 32: DRAW 2 AT AD,96: ROT= 0
442  IF FD = 2 THEN  GOSUB 230
445 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
450  HCOLOR= 4: DRAW 2 AT AF,57: DRAW 2 AT AG,57
460 AF = AF + S3:AG = AG + S3: IF AF > 279 THEN AF = 0
470  IF AG > 279 THEN AG = 0
480  HCOLOR= 7: DRAW 2 AT AF,57: DRAW 2 AT AG,57
482  IF FD = 3 THEN  GOSUB 230
485 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
490  HCOLOR= 4: ROT= 32: DRAW 2 AT AH,43: DRAW 2 AT AI,43
495 AH = AH - S4:AI = AI - S4: IF AH < 0 THEN AH = 279
500  IF AI < 0 THEN AI = 279
510  HCOLOR= 7: ROT= 32: DRAW 2 AT AH,43: DRAW 2 AT AI,43
530  HCOLOR= 1: ROT= FE: SCALE= 1
532  IF FD = 4 THEN  GOSUB 230
540  DRAW 1 AT FB,FC
550 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
560  POKE  - 16368,0
570  GOTO 370
2000  TEXT : HOME
2005  RESTORE
2010  VTAB 10
2020  PRINT "  SSSSS  PPPPPP  LL       AAAAA  TTTTTTT"
2030  PRINT " SSSSSSS PPPPPPP LL      AAAAAAA TTTTTTT"
2040  PRINT " SSS  SS PP   PP LL      AA   AA    TTT"
2050  PRINT " SSS     PP   PP LL      AA   AA    TTT"
2060  PRINT " SSSSSS  PPPPPPP LL      AAAAAAA    TTT"
2070  PRINT "  SSSSSS PPPPPP  LL      AAAAAAA    TTT"
2080  PRINT "     SSS PPP     LLLLLL AA   AA    TTT"
2090  PRINT " SSSSSSS PPP     LLLLLL AA   AA    TTT"
2100  PRINT "  SSSSS  PPP     LLLLLL AA   AA    TTT"
2110  FOR I = 1 TO 2
2112  FOR II = 1 TO 25: READ DD: POKE 0,DD: CALL 768: NEXT
2114  RESTORE : NEXT
2120 LL = LL + 1: IF LL > 3 THEN 5000
2130  RETURN
5000  TEXT : HOME
5020  PRINT : PRINT : PRINT "  BAD LUCK!"
5030  PRINT : PRINT "  YOURE SCORE WAS ";SC
5040  IF SC > 5000 THEN  PRINT "                    NOT A BAD SCORE"
5050  FOR I = 1 TO 25: READ DD: NEXT
5055  FOR I = 255 TO 1 STEP  - 3: POKE 0,I: CALL 768: NEXT
5060  DATA 250,250,250,250,200,200,200,150,150,150,100,100,50,0,0,0,0,200,20
0,0,0,0,200,200
5100  END
8000  POKE 768,169: POKE 769,4: POKE 770,133: POKE 771,1: POKE 772,234: POKE 773
,234: POKE 774,234: POKE 775,173: POKE 776,48:
8001  POKE 777,192: POKE 778,136: POKE 779,208: POKE 780,4: POKE 781,198: POKE 7
82,1: POKE 783,240: POKE 784,8: POKE 785,202:
8002  POKE 786,208: POKE 787,246: POKE 788,166: POKE 789,0: POKE 790,76: POKE 79
1,7: POKE 792,3: POKE 793,96: POKE 794,208:
8010  RETURN
10000  HCOLOR= 4: DRAW 1 AT FB,FC
10002  HCOLOR= 1: DRAW 1 AT FB,9
10005  PRINT  CHR$ (4);"BLOAD CHAIN,A520"
10010  CALL 520"FROGGER2"
19000  REM DATA FOR SHAPES
20000  DATA   2,0,6,0,40,0,63,54,63,36,60,54,54,46,53,54,63,54,54,37,44,45,45,45
,53,46,36,36,63
20005  DATA   36,44,37,36,36,55,54,63,36,63,0,63,63,39,36,63,63,63,55,54,63,63,5
4,54,54,54,54,54,54
20010  DATA   45,45,54,46,45,45,45,36,44,45,45,45,45,45,54,46,45,45,45,36,44,45,
36,37,44,36,37,60,36
20015  DATA   39,60,36,63,39,36,63,63,63,55,54,63,63,63,0,0,0,0,0
20018  FOR I = 1 TO 25: READ D: NEXT
20020  FOR LOC = 16384 TO 16485: READ PP: POKE LOC,PP: NEXT LOC
20030  RETURN
```

PERSPECTIVE

If you've ever wondered how they do those fancy 3D graphics displays on the TV ads, then this program is for you. The program requires the user to supply the 3D coordinates of the vertices which make up an object. The user must also specify which points are to be joined by straight lines. Once the object has been defined, it is displayed on the high resolution graphics screen and can be scaled up or down or rotated about any of the three spatial axes using a joystick connected to Port 2.

The data which defines the shape can also be saved on a tape file for subsequent reloading.

Objects can be described with up to 48 vertices, allowing quite realistic 'wire frame' drawings to be displayed. The display shows the object in true perspective, initially straight down the 'Y' axis. Rotation about the 'Z' axis is achieved by moving the joystick left or right, about the 'X' axis by moving the joystick up or down and about the 'Y' axis by pressing the fire button. With the fire button held down the object continues to rotate about the 'Y' axis. When the fire button is next pressed rotation recommences but in the opposite direction. Rotation about all three axes at once is thus possible.

The HIRES graphics routines must be present in memory before the program is run. The program itself is then loaded in two parts – a BASIC program which POKEs the perspective graphics routines and variables into memory, followed by the program which allows definition of shapes, scaling and tape file handling. The loader program splits the data into eleven separate blocks to facilitate checking and correction of data.

Shapes are rotated by increments of two degrees and the speed of rotation is dependant on the complexity (number of vertices) of the object. Whilst rapid, smooth movement is not possible with a computer such as the Commodore 64, movement is acceptably fast – but judge that for yourself.

Definition of shapes is accomplished in two stages:
1) The X, Y and Z coordinates of each of the vertices which make up the object are first supplied by the user. In general, these coordinates should not

```
10 POKE52,64:POKE56,64
20 FORX=0TO45
30 SX=SIN(X*π/90)*256
40 SX=INT(SX+.5)
50 IFSX>255THENSX=255
60 POKE16384+X,SX
70 NEXT
80 DIMX(48):DIMY(48):DIMZ(48):DIME(96)
150 PRINTCHR$(147):PRINT:PRINT"INPUT SHAPE FROM -"
160 PRINT:PRINT"1. TAPE FILE":PRINT"2. KEYBOARD":PRINT
170 INPUTM
180 IFM<1ORM>2THEN150
190 ONMGOSUB1000,2000
200 PRINTCHR$(147):PRINT:PRINT"LOADING SHAPE"
205 PRINT:PRINT"(PRESS F7 TO EXIT 3-D DISPLAY)"
210 POKE16430,NV
220 FORI=1TONV
240 POKE16431+I,ABS(X(I)):POKE16479+I,-255*(X(I)>=0)
250 POKE16527+I,ABS(Y(I)):POKE16575+I,-255*(Y(I)>=0)
260 POKE16623+I,ABS(Z(I)):POKE16671+I,-255*(Z(I)>=0)
270 NEXT
290 POKE16431,NE
300 FORI=1TONE
315 POKE16720+NE-I,ABS(E(I)):POKE16816+NE-I,-255*(E(I)<0)
320 NEXT
330 SYS17536
340 POKE53265,PEEK(53265)AND223:POKE53272,21
345 PRINTCHR$(147)
350 PRINT:PRINT"1. RE-RUN PRESENT SHAPE FROM START"
360 PRINT"2. SAVE PRESENT SHAPE ON TAPE FILE"
365 PRINT"3. RE-SCALE PRESENT SHAPE"
370 PRINT"4. INPUT NEW SHAPE"
380 PRINT"5. EXIT PROGRAM"
390 PRINT:INPUTM
400 IFM<1ORM>5THEN345
```

```
442  IF FD = 2 THEN  HCOLOR= 1: DRAW 1 AT FB,FC
444  GOSUB 230
445 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
450  HCOLOR= 4: DRAW 2 AT AF,52: DRAW 2 AT AG,52
460 AF = AF + S3:AG = AG + S3: IF AF > 279 THEN AF = 0
470  IF AG > 279 THEN AG = 0
480  HCOLOR= 7: DRAW 2 AT AF,52: DRAW 2 AT AG,52
481  IF FD = 3 THEN  HCOLOR= 4: DRAW 1 AT FB,FC:FB = FB + SF: IF FB > 270 OR FB
< 10 THEN 300
482  IF FD = 3 THEN  HCOLOR= 1: DRAW 1 AT FB,FC
484  GOSUB 230
485 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
490  HCOLOR= 4: DRAW 2 AT AH,25: DRAW 2 AT AI,25
495 AH = AH + S4:AI = AI + S4: IF AH < 0 THEN AH = 279
500  IF AI < 0 THEN AI = 279
510  HCOLOR= 7: DRAW 2 AT AH,25: DRAW 2 AT AI,25
530  HCOLOR= 1: ROT= FE: SCALE= 1
531  IF FD = 4 THEN  HCOLOR= 4: DRAW 1 AT FB,FC:FB = FB + SF: IF FB > 270 OR FB
< 10 THEN 300
532  IF FD = 4 THEN  HCOLOR= 1: DRAW 1 AT FB,FC
533  HCOLOR= 4: DRAW 1 AT FB,FC
534  GOSUB 230
537  IF FB > 270 OR FB < 10 THEN  GOTO 300
540  HCOLOR= 1: DRAW 1 AT FB,FC
550 X =  PEEK ( - 16384): IF X > 127 THEN  GOSUB 80
560  POKE  - 16368,0
570  GOTO 370
2000  TEXT : HOME
2005  RESTORE
2010  VTAB 10
2020  PRINT "  SSSSS  PPPPPP  LL       AAAAA  TTTTTTT"
2030  PRINT " SSSSSSS PPPPPPP LL      AAAAAAA TTTTTTT"
2040  PRINT " SSS  SS PP   PP LL      AA   AA    TTT"
2050  PRINT " SSS     PP   PP LL      AA   AA    TTT"
2060  PRINT " SSSSSS  PPPPPPP LL      AAAAAAA    TTT"
2070  PRINT "  SSSSSS PPPPPP  LL      AAAAAAA    TTT"
2080  PRINT "     SSS PPP     LLLLLL AA   AA    TTT"
2090  PRINT " SSSSSSS PPP     LLLLLL AA   AA    TTT"
2100  PRINT "  SSSSS  PPP     LLLLLL AA   AA    TTT"
2110  FOR I = 1 TO 2
2112  FOR II = 1 TO 30: READ DD: POKE 0,DD: CALL 768: NEXT
2114  RESTORE : NEXT
2115 SF = 0
2120 LL = LL + 1: IF LL > 3 THEN 5000
2130  RETURN
5000  TEXT : HOME
5020  PRINT : PRINT : PRINT "  BAD LUCK!"
5030  PRINT : PRINT "  YOURE SCORE WAS ";SC
5040  IF SC > 5000 THEN  PRINT "                  NOT A BAD SCORE"
5050  FOR I = 1 TO 25: READ DD
5055  FOR I = 0 TO 255 STEP 6: POKE 0,I: CALL 768: NEXT
5060  DATA 250,250,250,250,250,200,200,200,150,150,100,100,0,0,0,255,0,0,0,255,0
,0,0,255,0,0,0,0,0,0
5066  FOR I = 255 TO 1 STEP  - 6: POKE 0,I: CALL 768: NEXT
5100  END
7000  REM  FD=5, OR HOME!!
7010  IF FB < 50 AND FB > 30 AND H1 = 0 THEN H1 = 1: GOTO 7100
7020  IF FB < 100 AND FB > 80 AND H2 = 0 THEN H2 = 1: GOTO 7100
7030  IF FB < 150 AND FB > 130 AND H3 = 0 THEN H3 = 1: GOTO 7100
7040  IF FB < 200 AND FB > 180 AND H4 = 0 THEN H4 = 1: GOTO 7100
7050  IF FB < 250 AND FB > 230 AND H5 = 0 THEN H5 = 1: GOTO 7100
7060  GOTO 300
7100  REM  SUCCESS!!
7102  FOR I = 1 TO 12: POKE 0, INT ( RND (1) * 100 + 1): CALL 768: NEXT
7105 HH = 0
7110  IF H1 = 1 THEN  DRAW 1 AT 40,9:HH = HH + 1
7120  IF H2 = 1 THEN  DRAW 1 AT 90,9:HH = HH + 1
7130  IF H3 = 1 THEN  DRAW 1 AT 140,9:HH = HH + 1
7140  IF H4 = 1 THEN  DRAW 1 AT 190,9:HH = HH + 1
7150  IF H5 = 1 THEN  DRAW 1 AT 240,9:HH = HH + 1
7160  IF HH = 5 THEN H1 = 0:H2 = 0:H3 = 0:H4 = 0:H5 = 0:SC = SC + 1000: FOR I =
1 TO 30: POKE 0, INT ( RND (1) * 255): CALL 768: NEXT
7170  GOTO 10000
8000  POKE 768,169: POKE 769,4: POKE 770,133: POKE 771,1: POKE 772,234: POKE 773
,234: POKE 774,234: POKE 775,173: POKE 776,48:
8001  POKE 777,192: POKE 778,136: POKE 779,208: POKE 780,4: POKE 781,198: POKE 7
```

LINEAR EQUATION

Two co-ordinates are entered in the form (X1,Y1) and (X2,Y2). From this data, the computer will work out the equation of the line joining these two points. It takes into account whether the line is vertical or not.

In addition, the computer will also give the midpoint, distance, gradient, y-intercept and x-intercept of the line. It takes into account whether the gradient, y-intercept or x-intercept is undefined.

Great for working out maths homework. Will work on any computer using BASIC.

S. Chan
Minto Heights NSW

```
10 CLS
20 INPUT'ENTER CO-ORDINATES (X1,Y1)';X1,Y1
30 INPUT'ENTER CO-ORDINATES (X2,Y2)';X2,Y2
40 PRINT:PRINT:PRINT
50 REM  +MIDPOINT+
60 M1=(X1+X2) 2 : M2=(Y1+Y2)/2
70 PRINT'MIDPOINT     : (';M1;' ,';M2;' )'
80 REM  +DISTANCE+
90 D=SQR(((X1-X2).TM.(X1-X2))+((Y1-Y2).TM.(Y1-Y2)))
100  PRINT'DISTANCE     :';D
110 REM +GRADIENT+
120 IF X1-X2=0 THEN 160
130 G=(Y1-Y2)/(X1-X2)
140 PRINT'GRADIENT     :';G
150 GOTO 170
160 PRINT'GRADIENT     : LINE IS VERTICAL'
170 REM +Y-INTERCEPT+
180 IF X1-X2=0 THEN 220
185 IF Y1-Y2=0 THEN 195
190 B=Y1-(G.TM.X1):GOTO 200
195 B=Y1
200 PRINT'Y-INTERCEPT :';B
210 GOTO 230
220 PRINT'Y-INTERCEPT : UNDEFINED'
230 REM +X-INTERCEPT+
240 IF Y1-Y2=0 THEN 280
245 IF X1-X2=0 THEN 255
250 C=-B/G :GOTO 260
255 C=X1
260 PRINT'X-INTERCEPT :';C
270 GOTO 290
280 PRINT'X-INTERCEPT : UNDEFINED'
290 PRINT:PRINT
300 REM +EQUATION+
310 IF X1-X2=0 THEN PRINT'EQUATION   : X =';X1:GOTO 400
320 IF Y1-Y2=0 THEN PRINT'EQUATION   : Y =';Y1:GOTO 400
330 IF B=0 THEN PRINT'EQUATION   :  Y =';G;'X':GOTO 400
340 IF B.LT.0 THEN PRINT'EQUATION    : Y =';G;'X -';ABS(B):GOTO 4
00
350 PRINT'EQUATION   : Y =';G;'X +';B
400 REM
410 PRINT.AT.960,'PRESS 'L' TO RUN AGAIN';
420 IF INKEY$.LT..GT.'L' THEN 420 ELSE RUN
```

FROGGER

```
▷ 82,1: POKE 783,240: POKE 784,3: POKE 785,202:
3002  POKE 786,208: POKE 787,246: POKE 788,164: POKE 789,0: POKE 790,76: POKE 79
1,7: POKE 792,3: POKE 793,96: POKE 794,200:
3010  RETURN
10000  HCOLOR= 4: DRAW 1 AT 55,50
10005  PRINT  CHR$ (4);"BLOAD CHAIN,A520"
10010  CALL 520"FROGGER"
20000  DATA 3,0,6,0,10,0,63,54,63,36,60,54,54,48,53,54,63,54,54,37,44,45,45,45,5
3,46,36,36,63,
20005  DATA 36,44,37,36,36,55,54,63,36,63,0,63,63,63,55,55,55,55,54,55,54,55,54,
54,46,54,46,54,46,
20010  DATA 46,46,46,45,45,45,45,45,45,45,45,37,37,37,37,36,37,36,37,36,36,60
,36,60,36,60,60,60,
20015  DATA 60,63,63,63,63,63,63,63,0,63,55,54,63,63,63,
20018  FOR I = 1 TO 30: READ D: NEXT
20020  FOR I = 16384 TO 16484: READ D: POKE I,D: NEXT
20030  RETURN
```

SPECIAL FUNCTION KEYS

This program will let you type in commonly used DOS commands (CATALOG etc) and Applesoft reserved words (INPUT, FOR, NEXT etc) using the control characters. For example, typing control-I will cause the word INPUT to appear exactly as though you have just typed it in from the keyboard, character by character-but it only takes a small fraction of the time. Great news for hunt and peckers!

The list of keywords and the control characters which represent them are given in the table. Putting stickers on the keys is fine in the short term, but eventually they tend to gum up the works (pun intended).

Notice that not all the available control characters are used. Some are used by Apple for special purposes (namely control – C,D,G,H,J,K,M,S,U,X).

To type in this program, first ensure that DOS has been booted, then enter the Monitor by typing CALL –151 when you will see the prompt *. Now simply type in each line of the hex code as it appears in the listing – begin each line with the line number, to be followed immediately by a colon (:) and then the first 2 digit hex code and so on. After entering the program, type 3DOG to return to Applesoft. Save the program on disk using the command:
BSAVE CUSTOM KEYS, A$9500, L$FF

To use the control character utility program, simply BRUN it from disc after booting DOS. Better still, BRUN it in your HELLO program. This program can be temporarily disconnected by a RESET or a CALL 38164. When BRUN from disk this program will be located in memory just below DOS at starting address $9500 (hex). It also protects itself from being trampled upon by Applesoft by resetting HIMEM.

This program will work on an Apple II plus with DOS 3.3, an Apple work-a-like (provided it is sufficiently alike) or an Apple IIe in the 40 column mode. It is incompatible with the Apple IIe 80 column firmware which uses many of the control characters to provide special 80 column functions.

Derek Chan
Hawker ACT

CAST OF CONTROL CHARACTERS

COMMAND	KEY	HINT
CATALOG	control V	disc Volume
LIST	control L	List
RUN	control R	Run
FOR	control F	For
NEXT	control N	Next
STEP	control Z	Ztep
THEN	control T	Then
CALL	control A	cAll
PEEK(	control E	pEek(
POKE	control O	pOke
PLOT	control P	Plot
GOSUB	control B	gosuB
GO TO	control Y	Y looks like a branch
INPUT	control I	Input
PRINT	?	Applesoft treats ? as PRINT
Clrs scrn set 40 col	control W	Wipes screen Window

```
*9500.95FF
9500- A9 28 85 38 A9 95 85 39
9508- 20 EA 03 A9 FF 85 73 A9
9510- 94 85 74 60 A9 1B 85 38
9518- A9 FD 85 39 20 EA 03 60
9520- 9D 00 02 E8 20 F0 FD 60
9528- 20 1B FD 18 C9 9B 90 01
9530- 60 C9 97 D0 0A A9 28 85
9538- 21 20 58 FC A9 A0 60 C9
9540- B0 F0 FB C9 B3 F0 F7 C9
9548- B4 F0 F3 C9 B7 F0 EF C9
9550- B8 F0 EB C9 BA F0 E7 C9
9558- BB F0 E3 C9 BD F0 DF C9
9560- 91 F0 DB C9 93 F0 D7 C9
9568- 95 F0 D3 C9 9B F0 CF 38
9570- E9 80 8D 9D 95 A0 00 B9
9578- 9E 95 C8 C9 AA F0 03 4C
9580- 77 95 CE 9D 95 AD 9D 95
9588- C9 00 F0 03 4C 77 95 B9
9590- 9E 95 C9 AA F0 A6 20 20
9598- 95 C8 4C BF 95 00 AA C3
95A0- C1 CC CC AA C7 CF D3 D5
95A8- C2 AA AA AA D0 C5 C5 CB
95B0- A8 AA C6 CF D2 AA AA AA
95B8- C9 CE D0 D5 D4 AA AA AA
95C0- CC C9 D3 D4 AA AA CE C5
95C8- D8 D4 AA D0 CF CB C5 AA
95D0- D0 CC CF D4 AA AA D2 D5
95D8- CE AA AA D4 C8 C5 CE AA
95E0- AA C3 C1 D4 C1 CC CF C7
95E8- AA AA AA C7 CF A0 D4 CF
95F0- AA D3 D4 C5 D0 AA D5 D4
95F8- AA AA C3 C1 D4 C1 CC 00
*
```

CATALOG INTERRUPT

```
10  REM
20  REM      CATALOGUE INTERRUPT BY
30  REM        D.S.YAN, 1984
40  REM
50  REM  ************************************************************
60  REM  * Change DOS to return to BASIC after printing 18 lines of catalog *
90  REM  ************************************************************
95  REM
100   POKE 44601,76: POKE 44602,127: POKE 44603,179
110   REM
120   REM  ***********************************************
130   REM  * Change DOS to jump to $318 on end of catalog *
150   REM  ***********************************************
160   REM
170   POKE 44589,24: POKE 44590,3
180   REM
190   REM  ************************************************************
200   REM  * Set up code at $310 to store stack pointer at $B39B then enter *
230   REM  * catalog routine at $AE25                                    *
250   REM  ************************************************************
260   REM
270   FOR I = 784 TO 791: READ J: POKE I,J: NEXT
280   REM
290   REM  ***********************************************************
300   REM  * Set up code at $318 to clear flag at end  of directory *
330   REM  ***********************************************************
340   REM
350   FOR I = 792 TO 800: READ J: POKE I,J: NEXT
355   HOME
360   D$ =  CHR$ (4): REM   <CTRL-D>
370   REM
380   REM  ****************************
390   REM  * Print first catalog page *
400   REM  ****************************
410   REM
420   PRINT D$;"CATALOG"
430   REM
440   REM  ********************************
450   REM  * Print instructions on screen *
470   REM  ********************************
480   REM
490   VTAB 1: HTAB 1: INVERSE : PRINT " <- UP / -> DOWN / <RETURN>
       NEXT PAGE": NORMAL
500   VTAB 24: INVERSE : PRINT "<D>DELETE<U>UNLOCK<L>LOCK<R>RUN<O>LOAD"
      ;: NORMAL
502   REM
504   REM  ************************************************
505   REM  * Reduce TEXT window by one line, top and bottom *
506   REM  ************************************************
507   REM
510   POKE 34,1: POKE 35,23
520   CD = 5: HTAB 8
530   IF CD < 2 THEN CD = CD + 1: GOTO 550
540   IF CD > 23 THEN CD = CD - 1
550   VTAB (CD)
560   GET CU$
570   REM
580   REM  ***************
590   REM  * Move cursor *
600   REM  ***************
610   REM
620   IF  ASC (CU$) = 08 THEN CD = CD - 1: GOTO 530
630   IF  ASC (CU$) = 21 THEN CD = CD + 1: GOTO 530
640   REM
650   REM  ***************************************
660   REM  * GOTO next catalog page on <RETURN> *
680   REM  ***************************************
690   REM
700   IF  ASC (CU$) = 13 THEN 1120
710   REM
720   REM  ****************************
730   REM  * Exit programme on <ESC> *
740   REM  ****************************
750   REM
760   IF  ASC (CU$) = 27 THEN 1350
770   IF  ASC (CU$) <  > 68 AND  ASC (CU$) <  > 76 AND  ASC (CU$)
      <  > 82 AND  ASC (CU$) <  > 85 AND  ASC (CU$) <  > 79 THEN 560
780   GOSUB 940
```

```
790  REM
800  REM  ****************************
810  REM  * Read filename from screen *
820  REM  ****************************
830  REM
840   PK = PK - 1:PK$ =  CHR$ ( PEEK (PK))
850   PK = PK + 1
853  REM
854  REM  ************************************************************
855  REM  * Check for five consecutive spaces on screen to indicate *
857  REM  * end of filename                                          *
858  REM  ************************************************************
859  REM
860   IF  PEEK (PK) = 160 AND  PEEK (PK + 1) = 160 AND  PEEK (PK + 2)
     = 160 AND  PEEK (PK + 3) = 160 AND  PEEK (PK + 4) = 160 THEN 980
870   PK$ = PK$ +  CHR$ ( PEEK (PK))
880   GOTO 850
890  REM
900  REM  ************************************************
910  REM  * Locate memory address of cursor screen position *
920  REM  ************************************************
930  REM
940   IF CD > 0 AND CD < 9 THEN PK = 1031 + 128 * (CD - 1)
950   IF CD > 8 AND CD < 17 THEN PK = 1071 + 128 * (CD - 9)
960   IF CD > 16 AND CD < 25 THEN PK = 1111 + 128 * (CD - 17)
970   FT =  PEEK (PK - 6): RETURN
980   PRINT D$
990   IF CU$ = "D" THEN  PRINT D$;"DELETE";PK$: VTAB CD: HTAB 1:
      CALL  - 868: VTAB CD: HTAB 8
1000  IF CU$ = "L" THEN  PRINT D$;"LOCK";PK$: VTAB CD: HTAB 1:
      PRINT "*": VTAB CD: HTAB 8
1010  IF CU$ = "R" OR CU$ = "O" THEN 1040
1020  IF CU$ = "U" THEN  PRINT D$;"UNLOCK";PK$: VTAB CD: HTAB 1:
      PRINT " ": VTAB (CD): HTAB 8
1030  GOTO 560
1040  IF FT = 212 THEN  VTAB 2: HTAB 1: CALL  - 868: HTAB 10: FLASH
      : PRINT "FILE TYPE MISMATCH": NORMAL : HTAB 8: VTAB CD: GOTO 560
1050  GOTO 1350
1060 REM  ********************************************
1070 REM  * Set up DOS to read next 21 catalog entries *
1100 REM  ********************************************
1110 REM
1120  POKE 45981,21
1130 REM
1140 REM  ************************************************************
1150 REM  * Check catalog end flag at $320 and exit if equal to zero *
1180 REM  ************************************************************
1190 REM
1200  IF  PEEK (800) = 0 THEN 1350
1210 REM
1220 REM  ***********************
1230 REM  * GOTO routine at $310 *
1240 REM  ***********************
1250 REM
1260  VTAB 23: HTAB 1
1270  CALL 784
1280  CD = 2: HTAB 8: GOTO 530
1290 REM
1300 REM  ****************************
1310 REM  * Change DOS back to normal *
1330 REM  ****************************
1340 REM
1350  POKE 44601,32: POKE 44602,12: POKE 44603,253
1360  POKE 44589,127: POKE 44590,179
1370  POKE 34,0: POKE 35,24
1380  HOME
1390 REM
1400 REM  ********************************************
1410 REM  * RUN,BRUN,LOAD or BLOAD filename as selected *
1430 REM  ********************************************
1440 REM
1450  IF FT = 193 AND CU$ = "R" THEN  PRINT D$;"RUN";PK$
1460  IF FT = 193 AND CU$ = "O" THEN  PRINT D$;"LOAD";PK$
1470  IF FT = 194 AND CU$ = "R" THEN  PRINT D$;"BRUN";PK$
1480  IF FT = 194 THEN  PRINT D$;"BLOAD";PK$
1490  END
1500  DATA   186,142,155,179,32,37,174,96,169,0,141,32,3,76,127,179,1
```

Writing programs requiring retrieval of files from a disk, I found it desirable to have the catalog of disk files displayed on the screen from within a BASIC program to assist the program user to enter the filename as it is stored on the disk. A more useful feature would be to have the displayed file selectable.

The program is essentially in two parts. The first part handles the DOS alteration to display one 'page' of the catalog at a time (18-21 lines). The second part of the program manipulates the screen cursor and performs DOS commands on filenames present on the screen.

The program listing contains a liberal sprinkling of REMark statements which summarise the workings of the program. These lines can be left out when typing in the program, as they are not referenced by any GOTO's or GOSUB's. A list of the main variables used are given in figure 1.

Note: It is advisable when first running the program to use a backup copy of your disk and double check the expressions in the POKE statements. Indiscriminate poking around in DOS could produce disastrous results.

Denis Yan
Ingleburn NSW

```
5  DIM J$(100)
10  GOTO 360
20  POKE 216,0
25  RESTORE
30  FOR X = 1 TO 6: READ B$(X): NEXT

50  DATA  LOAD,LOCK,UNLOCK,DELETE
    ,RENAME,EXEC
60  TEXT : HOME :D$ =  CHR$ (4): PRINT
    D$"CATALOG":B =  PEEK (37) -
    2: IF B > 22 THEN B = 22
70 T = 0:CH = 4: FOR CV = 0 TO 23
    : GOSUB 260: IF C <  > 160 THEN
     POKE P - 1,219: POKE P,T +
    193: POKE P + 1,221:T = T +
    1:S = CV
80  NEXT CV: VTAB 24:A$ = "TYPE L
    ETTER TO RUN, OR     LOAD =
    1     LOCK = 2     UNLOCK =
    3     DELETE = 4     RENAME
    = 5     EXEC = 6     SYS. GE
    N. = 7     FLASH - CATALOG =
     8     EXIT = 9     .......
       "
90 B$ = "RUN": HTAB 1: PRINT  LEFT$
    (A$,39);:A$ =  MID$ (A$,2) +
     LEFT$ (A$,1):K =  PEEK ( -
    16384): IF K < 128 THEN  FOR
    K = 1 TO 75: NEXT K:K =  FRE
    (0): GOTO 90
100  POKE  - 16368,0:K = K - 176:
     IF K < 0 OR K > 9 THEN 200
105  IF K = 9 THEN  NEW
110  IF K = 7 THEN 400
115  IF K = 8 THEN 800
120  IF K > 9 THEN 90
130  HTAB 1: CALL  - 868: IF K =
    0 THEN  NEW
140  PRINT "PRESS LETTER YOU WISH
     TO ";
150  IF K = 4 THEN  FLASH
160  PRINT B$(K);: NORMAL
170 B$ = B$(K)
180  ONERR  GOTO 290
190  CALL  - 198: NORMAL : GET K$
    :K =  ASC (K$) - 48
200  IF K < 17 OR K > T + 16 THEN
    : HOME : CLEAR : PRINT  CHR$
    (4);"RUN HELLO"
210 CH = 1:CV = S - T + K - 16: GOSUB
    260: IF C = 194 AND (B$ = "R
    UN" OR B$ = "LOAD") THEN B$ =
    "B" + B$
220  FOR CH = 6 TO 39: GOSUB 260:
    B$ = B$ +  CHR$ (C): NEXT CH
    : IF  LEFT$ (B$,6) = "RENAME
    " THEN 280
230  VTAB  PEEK (37) + 1: HTAB 1:
     CALL  - 868: PRINT B$: PRINT
    D$;B$
240  IF  LEFT$ (B$,4) = "EXEC" THEN
     HOME : PRINT D$;"MON C,I,O"
    : END
250  GOTO 30
260 C1 =  INT (CV / 8):C2 = CV -
    C1 * 8:P = 1024 + 128 * C2 +
    40 * C1 + CH:C =  PEEK (P): RETUR
270  FOR CH = 6 TO 39: GOSUB 260:
    B$ = B$ +  CHR$ (C): NEXT CH
    : HTAB 1: CALL  - 868: PRINT
    B$: PRINT D$;B$: GOTO 30
280  HTAB 1: VTAB 23: PRINT "ORIG
    INAL "; MID$ (B$,7): VTAB  PEEK
```

```
(372 HTAB 12 CALL  - 868: HTAB
1: INPUT "NEW FILENAME ? ";N
E$: PRINT  CHR$ (4);B$; CHR$
(44);NE$: GOTO 30
290 POKE 216,0:ERR =  PEEK (222)
: IF  LEFT$ (B$,6) = "DELETE
" THEN 340
300 IF ERR = 10 THEN  VTAB 23: PRINT
"FILE LOCKED: (C)ONTINUE OR
 (A)BORT ";
310 GET N$
320 IF N$ = "C" THEN  PRINT  CHR$
(4);"UNLOCK"; MID$ (B$,7): PRINT
 CHR$ (4);B$;",";NE$
330 GOTO 30
340 IF ERR = 10 THEN  HTAB 1: VTAB
23: PRINT "FILE LOCKED: (C)O
NTINUE OR (A)BORT ";: GET N$
: IF N$ = "C" THEN  PRINT D$
;D$;"UNLOCK"; MID$ (B$,7): PRINT
D$;B$: GOTO 30
350 GOTO 30
360 ONERR  GOTO 390
370 PRINT  CHR$ (4);"EXEC STARTU
P FILE"
380 END
390 GOTO 20
400 HOME : INVERSE : PRINT "SYST
EM GENERATOR"
410 PRINT  CHR$ (7)
420 NORMAL : PRINT : PRINT "1 -
GENERATE NEW HELLO PROGRAM"
430 PRINT : PRINT "2 - GENERATE
STARTUP FILE"
440 PRINT : PRINT "3 - INITIALIS
E DISK"
450 PRINT : PRINT "4 - GENERATE
EXEC FILE"
460 PRINT : PRINT "X - EXIT TO M
AIN MENU"
470 GET A$: PRINT
472 IF A$ = "1" THEN F$ = "": GOTO
750
475 IF A$ = "3" THEN 690
480 IF A$ = "4" THEN 620
490 IF A$ = "X" THEN  RUN
500 IF A$ <  > "2" THEN 400
510 HOME : INVERSE : PRINT "STAR
TUP FILE GENERATOR": NORMAL
: PRINT
520 GOSUB 660
530 PRINT :F$ = "STARTUP FILE":X
 = 0
540 INPUT J$(X): IF J$(X) = "" THEN
570
550 IF J$(X) =  CHR$ (2) AND X >
0 THEN X = X - 1: PRINT J$(X
): PRINT "ENTER FROM HERE ON
WARDS": PRINT : GOTO 540
560 X = X + 1: IF X < 99 THEN 540
570 PRINT D$;"OPEN ";F$: PRINT D
$;"DELETE ";F$
580 PRINT D$;"OPEN ";F$: PRINT D
$;"WRITE ";F$
590 FOR J = 0 TO X: PRINT J$(J):
 NEXT J
600 PRINT D$;"CLOSE ";F$
610 PRINT : PRINT "DONE.  PRESS
ANY KEY TO CONTINUE...";: GET
A$: GOTO 400
620 PRINT : PRINT
630 INPUT "WHAT FILENAME ? ";F$
640 IF  VAL (F$) <  > 0 OR F$ =
"" THEN  PRINT  CHR$ (7)"ILL
```

'ULLO 'ULLO

This program includes two new features:

1. The 'Flash Catalog' routine from the Apple DOS (3.3) Manual, which displays hidden control characters in file names as flashing letters*
2. The capability to create and use Exec files, and to set up a series of commands to be executed automatically on start-up.

These facilities may be operated by options 7 (Sys Gen) and 8 (Flash Catalog) on the menu; i.e: the message that scrolls around at the bottom of the screen. Files may be executed by using option 6 (Exec).

for the uninitiated, control characters can be put into file names as a sort of password; they can be extremely annoying if you forget what or where they are.

Andrew Maizels
Mt. Colah NSW

'ULLO 'ULLO

```
▷ EGAL FILENAME": PRINT : GOTO
   630
650 GOSUB 660: GOTO 540
660 HOME
670 PRINT "ENTER THE STATEMENTS
   YOU WISH TO BE     EXECUTED.
     PRESS RETURN BY ITSELF WHE
   N  YOU HAVE FINISHED.  USE C
   TRL-B <RETURN>  TO CHANGE YO
   UR PREVIOUS ENTRY"
680 RETURN
690 HOME : PRINT  CHR$ (7): INPUT
   "INSERT DISK TO BE INITIALIS
   ED, THEN     PRESS RETURN...
   ";Z$
700 PRINT : PRINT : INPUT "WHAT
   FILENAME DO YOU WISH TO BE E
   XECUTEDON STARTUP ? ";F$
710 IF  VAL (F$) <  > 0 THEN  PRINT
    CHR$ (7);"ILLEGAL FILENAME"
   : PRINT : GOTO 700
715 PRINT : INPUT "WHAT VOLUME N
   UMBER DO YOU WANT ? ";A
720 PRINT : PRINT : INPUT "INSER
   T DISK TO BE INITIALISED, TH
   EN  PRESS RETURN...";Z$
730 PRINT D$;"INIT ";F$;",V";A
740 PRINT : PRINT "DO YOU WANT A
    COPY OF THIS PROGRAM ON  T
   HE DISK ? ": GET C$: IF Z$ =
   "N" THEN 400
750 HOME : INVERSE  PRINT "GENE
   RATING GREETINGS PROGRAM..."
   : NORMAL
760 IF F$ = "" THEN  PRINT : INPUT
   "WHAT FILENAME ? ";F$
770 PRINT D$;"SAVE ";F$
780 PRINT : PRINT "PRESS ANY KEY
    TO RETURN...": GET A$: GOTO
   400
800 HOME
805 RESTORE : FOR X = 1 TO 6: READ
   A$: NEXT
810 DATA    201,141,240,21,201,1
   36
820 DATA    240,17,201,128,144,1
   3
830 DATA    201,160,176,9,72,132

840 DATA    53,56,233,64,76,249
850 DATA    253,76,240,253
860 FOR I = 768 TO 768 + 27
870 READ V: POKE I,V: NEXT I
880 POKE 54,0: POKE 55,3
890 CALL 1002
900 PRINT "FLASH - CATALOG INSTA
   LLED AND READY.": PRINT : PRINT
   "PRESS ANY KEY TO CONTINUE..
   .": GET A$: GOTO 20
```

■

APPLE SPACE WAR

This is basically a Space Invaders type of game, with both player and aliens using laser weapons rather than missiles. The player has only one life, but starts with 100 energy points which decrease when he fires at, or is hit by, aliens.

It is possible to earn a score of 200, and 20 energy points, by hitting the strongest alien, but that being can inflict up to 105 points of damage on you!

The program includes instructions for playing, which may be chosen from the startup menu, and has two special features:

1 – The top fifteen scores, and the players names, are stored permanently on disk.

2 – A "Demo" mode, in which the computer plays both sides. If left unattended, the program will automatically enter this mode returning to the menu after each game to give a human a chance (if one is present). The computers top score is 180.

The game normally starts with one alien, with more (up to 20) appearing as the game progresses. The starting number may be altered by changing the value given to NT in line 490.

If you experience problems with the disk file, try changing line 1190 to read: 1190 PRINT D$;D$;"OPEN";F$. On the subject of the disk file, use the program "Hiscore Creator" to set up the file before your first game (or to erase the high score table later).

Andrew Maizels
Mt. Colah NSW

```
10  ONERR  GOTO 1260
20  DIM K(3)
30 K(0) = 8:K(1) = 21:K(2) = 32
40 DE$ = "D"
50  DIM D(21),C(21),T(21),A(21)
60  HOME : GOSUB 70: GOTO 330
70  PRINT "    ------------------
    ---
80  PRINT
90  PRINT "      APPLE SPACE WAR!
    !
100  PRINT
110  PRINT "      BY ANDREW MAIZE
    LS
120  PRINT "      COPYRIGHT 13.7.
    83
130  PRINT
140  PRINT "      ANOTHER GREAT G
    AME
150  PRINT "            FROM
160  PRINT "      GANYMEDE SOFTWA
    RE
170  PRINT
180  PRINT "      PHONE:(02) 477-
    2108
190  PRINT
200  PRINT "    ----------------
    -----
210  PRINT : PRINT : PRINT "PRESS
     ANY KEY TO PLAY, OR:"
220  PRINT : PRINT "D - DEMONSTRA
    TION GAME"
230  PRINT : PRINT "I - INSTRUCTI
    ONS"
240  PRINT : PRINT "X - EXIT"
250  POKE  - 16368,0:A =  PEEK ( -
    16384): IF A < 128 THEN B =
    B + 1: IF B < 500 THEN 250
260 A = A - 128
270  IF A = 88 THEN  END
280  IF A = 68 THEN DE$ = "D": GOT
    330
290  IF A = 73 THEN 1300
300  FOR X = 1 TO 75: NEXT
310  IF A < 0 THEN DE$ = "D": GOTO
    330
320 DE$ = ""
330  DIM NM$(16),SC(16)
340  GOTO 1170
350  GOSUB 380
360  GOTO 430
370  NEXT
380  GR : HOME : PRINT "---------
    --APPLE SPACE WAR!!---------
    --"
390  PRINT "----------BY ANDREW
    MAIZELS----------"
400  PRINT "HI SCORE:";NM$(1);"
    ";SC(1)
410  POKE 34,23
420  RETURN
430 S = 0:E = 0
440  HOME
450  COLOR= 2
460  FOR Y = 38 TO 39: HLIN 0,39 AT
     Y: NEXT
470 A = 20:X = 0:T = 1:C = 1:F =
    1
```

APPLE SPACE WAR

```
480  FOR X = 1 TO 20:D(X) =  INT
     ( RND (1) * 15) + 1:C(X) =  INT
     ( RND (1) * 29):T(X) = 1 +  INT
     ( LOG (X) + ( RND (1) * X)):
     A(X) =  INT ( RND (1) * 39):
     EN(X) = X * 3: NEXT
490 NT = 1
500 S% =  - 16336
510  FOR QV = 1 TO NT:A = A(QV):C
     = C(QV):T = T(QV)
520  COLOR= 0: PLOT A,C: PLOT A,C
     + 1:A = A + T
530  IF  INT ( RND (1) * 40) = 1 THEN
     T =  - T
540  IF A > 38 THEN A =  RND (1) *
     3
550  IF A < 0 THEN A = 39
560  IF C < 1 THEN C = 3
570 C = C + F: IF C < 1 OR C > 30
     THEN F =  - F
580 B% =  PEEK (S%)
590  COLOR= D(QV): PLOT A,C: PLOT
     A,C + 1
600 A(QV) = A:C(QV) = C:T(QV) = T
     : NEXT QV
610  PRINT "SCORE - ";S,"ENERGY -
     ";100 - E;"  ";
620  HTAB 1
630  IF E > 100 THEN 1260
640 K =  PEEK ( - 16384) - 128: POKE
     - 16368,0
650  IF DE$ = "D" THEN K = K( INT
     ( RND (1) * 3))
660 TT = TT + 1: IF TT > 100 THEN
     NT = NT + 1:TT = 0: IF NT >
     20 THEN NT = 20
670 MK = X
680  IF K = 8 THEN X = X - 3
690  IF K = 21 THEN X = X + 3
700  IF K = 32 THEN 850
710  IF X < 0 THEN X = 39
720  IF X > 39 THEN X = 0
730  COLOR= 0: PLOT MK,36: PLOT M
     K,37
740  COLOR= 12: PLOT X,36: PLOT X
     ,37
750  FOR QV = 1 TO NT:A = A(QV):C
     = C(QV)
760 I =  INT ( RND (1) * 10)
770  IF I <  > 1 THEN 830
780  COLOR= 13: VLIN C + 2,37 AT
     A
790  FOR NN = 1 TO 50: NEXT
800 B% =  PEEK (S%) +  PEEK (S%) -
     PEEK (S%) +  PEEK (S%) +  PEEK
     (S%) -  PEEK (S%)
810  COLOR= 0: VLIN C + 2,37 AT A

820  IF  INT (A) > X - 2 AND  INT
     (A) < X + 2 THEN  PRINT  CHR$
     (7); CHR$ (7); CHR$ (7):E =
     E + 5 + 5 * QV: IF E > 100 THEN
     920
830 A(QV) = A:C(QV) = C: NEXT QV
840  GOTO 510
850  COLOR= 15: VLIN 35,0 AT X: FOR
     NN = 1 TO 20: NEXT : COLOR=
     0: VLIN 35,0 AT X
860 B% =  PEEK (S%) +  PEEK (S%) -
     PEEK (S%) +  PEEK (S%) +  PEEK
     (S%) -  PEEK (S%) +  PEEK (S
     %) +  PEEK (S%) -  PEEK (S%)
     +  PEEK (S%)
870 E = E + 1
880  FOR Y = 1 TO NT
```

```
890  IF X =  INT (A(Y)) THEN S =
     S + 10 * Y:E = E - Y: PRINT
      CHR$ (7):
900  NEXT
910  GOTO 750
920  TEXT : HOME : PRINT "-------
     ------ HIGH SCORES ---------
     -----"
930  HTAB 1: PRINT "NAME:";: HTAB
     30: PRINT "SCORE:"
940  PRINT : PRINT
950  FOR X = 1 TO 15: VTAB X + 4:
      HTAB 1: PRINT NM$(X);: HTAB
     30: PRINT SC(X): NEXT
960  IF S > SC(15) THEN 1030
970  POKE  - 16368,0
980  IF DE$ = "D" THEN  FOR X = 1
      TO 5000: NEXT : RUN
990  POKE  - 16368,0: PRINT : PRINT
     "PRESS ANY KEY FOR ANOTHER G
     AME, OR 'X'  TO EXIT...";: GET
     A$: IF A$ = "X" THEN  END
1000  IF A$ = "D" THEN  RUN
1010 DE$ = "": GOSUB 380: FOR X =
     1 TO 1500: NEXT
1020  GOTO 430
1030  IF DE$ = "D" THEN NM$(15) =
     "THE COMPUTER": GOTO 1050
1040  PRINT : PRINT : INPUT "WHAT
      IS YOUR NAME, OH CHAMPION ?
     ";NM$(15)
1050  IF  LEN (NM$(15)) > 20 THEN
     NM$(15) =  LEFT$ (NM$(15),20
     )
1060 Y = 0
1070 SC(15) = S
1080  FOR X = 1 TO 14: IF SC(X) <
     SC(X + 1) THEN S = SC(X):SC(
     X) = SC(X + 1):SC(X + 1) = S
     :NM$ = NM$(X + 1):NM$(X + 1)
      = NM$(X):NM$(X) = NM$:Y = 1
1090  NEXT
1100  IF Y = 1 THEN Y = 0: GOTO 1
     080
1110 S =  - 1
1120 F$ = "HISCORE":D$ =  CHR$ (4
     )
1130  PRINT D$"OPEN "F$: PRINT D$
     "WRITE"F$
1140  FOR X = 1 TO 16: PRINT NM$(
     X): PRINT SC(X): NEXT
1150  PRINT D$"CLOSE"
1160  GOTO 920
1170 D$ =  CHR$ (4):F$ = "HISCORE
     "
1180  PRINT D$
1190  PRINT D$"OPEN"F$
1200  PRINT D$"READ"F$
1210  FOR X = 1 TO 15
1220  INPUT NM$(X),SC(X)
1230  NEXT
1240  PRINT D$"CLOSE"F$
1250  GOTO 350
1260  HOME : INVERSE : PRINT "GAM
     E OVER...";: NORMAL
1270  PRINT "ENERGY - ";
1280  FLASH : PRINT 100 - E:: NORMAL
1290  FOR X = 1 TO 800:A =  PEEK
     ( - 16336): NEXT : GOTO 920
1300  REM  INSTRUCTIONS
1310  HOME : INVERSE : PRINT  CHR$
     (7)"----------- APPLE SPACE
     WAR! -----------"
1320  PRINT  CHR$ (7)
1330  NORMAL : PRINT "    USE THE
      LEFT AND RIGHT ARROWS TO
         MOVE LEFT AND RIGHT."
1340  PRINT : PRINT "    USE THE
     SPACE BAR TO FIRE"
1350  PRINT   PRINT "    YOUR AIM
      IS TO SHOOT AS MANY OF THE
         ALIEN SPACE CRAFT ( COLO
     URED BLOBS )    AS YOU CAN,
     WHILE AVOIDING THEIR
     FIRE."
1360  PRINT : PRINT : PRINT "PRES
     S ANY KEY TO RETURN..."
1370  POKE  - 16368,0: GET A$: RUN
```

Hi-Score Creator for Apple Space War

```
10 F$="HISCORE" : D$=CHR$(4)
20 PRINT D$;"OPEN ";F$ : PRINT D$;"DELETE";F$
30 DIM NM$(15),SC(15)
40 FOR X = 1 TO 15 : NM$(X)="----------------" : NEXT X
50 PRINT D$;"OPEN";F$ : PRINT D$;"WRITE";F$
60 FOR X = 1 TO 15
70 PRINT NM$(X) : PRINT SC(X)
80 NEXT X
90 PRINT D$;"CLOSE";F$
```

SORTS

*A SORT FOR APPLE

APPLE STRING ARRAYS

N$	DIMENSIONS	L0	P,B	P,A	L1	P,B	P,A	L2	.....

L0 - Length N$(0)
P,B, P,A; address where string found.

```
]
SUPSORT      8/12/'83
]
] 00 01 EF D8 00
*
0FA0-   AD 50 10    LDA   $1050
0FA3-   8D 4A 10    STA   $104A
0FA6-   AD 51 10    LDA   $1051
0FA9-   8D 4B 10    STA   $104B
0FAC-   A2 02       LDX   #$02
0FAE-   8E F0 0F    STX   $0FF0
0FB1-   A2 FA       LDX   #$FA
0FB3-   A0 03       LDY   #$03
0FB5-   EE 4A 10    INC   $104A
0FB8-   AD 4A 10    LDA   $104A
0FBB-   D0 03       BNE   $0FC0
0FBD-   EE 4B 10    INC   $104B
0FC0-   88          DEY
0FC1-   C0 00       CPY   #$00
0FC3-   D0 F0       BNE   $0FB5
0FC5-   CA          DEX
0FC6-   E0 00       CPX   #$00
0FC8-   D0 E9       BNE   $0FB3
0FCA-   CE F0 0F    DEC   $0FF0
0FCD-   AD F0 0F    LDA   $0FF0
0FD0-   D0 DF       BNE   $0FB1
0FD2-   A9 00       LDA   #$00
0FD4-   8D 62 10    STA   $1062
0FD7-   4C 64 10    JMP   $1064
0FDA-   00          BRK
0FDB-   00          BRK
0FDC-   00          BRK
0FDD-   00          BRK
0FDE-   A2 00       LDX   #$00
0FE0-   BD 44 10    LDA   $1044,X
0FE3-   20 ED FD    JSR   $FDED
0FE6-   E8          INX
0FE7-   E0 06       CPX   #$06
0FE9-   D0 F5       BNE   $0FE0
0FEB-   60          RTS
0FEC-   00          BRK
0FED-   00          BRK
0FEE-   00          BRK
0FEF-   00          BRK
0FF0-   7F          ???
0FF1-   00          BRK
0FF2-   00          BRK
0FF3-   00          BRK
0FF4-   00          BRK
0FF5-   00          BRK
0FF6-   A5 6B       LDA   $6B
0FF8-   8D 01 10    STA   $1001
0FFB-   A5 6C       LDA   $6C
0FFD-   8D 02 10    STA   $1002
1000-   AD 40 12    LDA   $1240
1003-   C9 4E       CMP   #$4E
1005-   F0 15       BEQ   $101C
1007-   EE 01 10    INC   $1001
100A-   AD 01 10    LDA   $1001
100D-   D0 F1       BNE   $1000
100F-   EE 02 10    INC   $1002
```

GET ADDRESS FOR N$(500)
(This is one beyond the last item.)
Advance through lengths & pointers of string arrays 500 x 3 positions. (2*$FA*3 or 2 x 250$_{10}$ x 3).
The address for length of N$(500) is placed in 104A - 104B.

Place 0 in 1062 (non zero means adjacent items are being compared.)

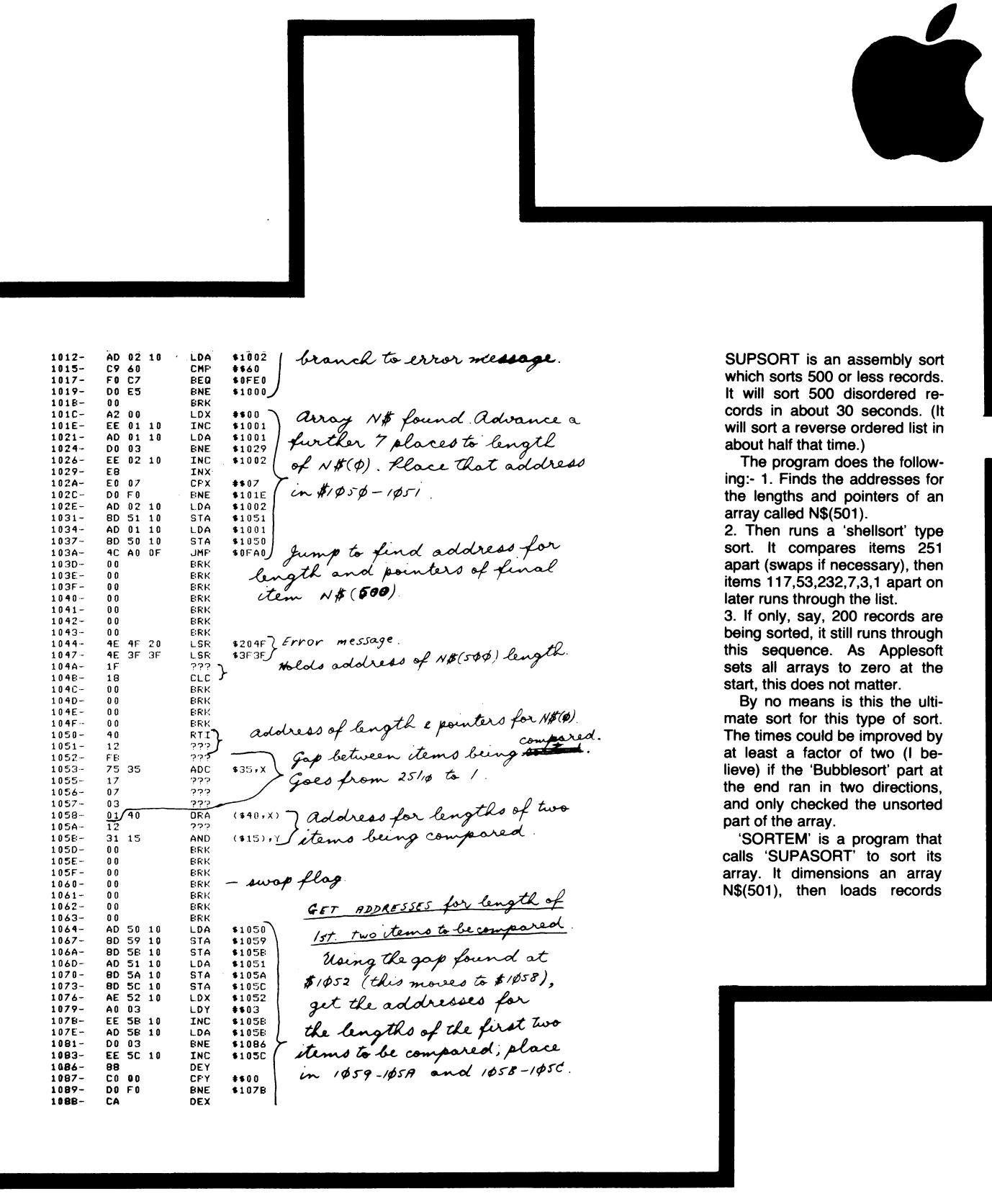
Output error message held at $1044 to $1049

Error message is a waste of time. Start of array should be found just above LOMEM.

START

6B & 6C hold address for start of array space.

Find "N" (ASCII 4E) which is start of N$ string array.
If not found before $6000.



```
1012-  AD 02 10   LDA  $1002
1015-  C9 60      CMP  #$60
1017-  F0 C7      BEQ  $0FE0
1019-  D0 E5      BNE  $1000
101B-  00         BRK
101C-  A2 00      LDX  #$00
101E-  EE 01 10   INC  $1001
1021-  AD 01 10   LDA  $1001
1024-  D0 03      BNE  $1029
1026-  EE 02 10   INC  $1002
1029-  E8         INX
102A-  E0 07      CPX  #$07
102C-  D0 F0      BNE  $101E
102E-  AD 02 10   LDA  $1002
1031-  8D 51 10   STA  $1051
1034-  AD 01 10   LDA  $1001
1037-  8D 50 10   STA  $1050
103A-  4C A0 0F   JMP  $0FA0
103D-  00         BRK
103E-  00         BRK
103F-  00         BRK
1040-  00         BRK
1041-  00         BRK
1042-  00         BRK
1043-  00         BRK
1044-  4E 4F 20   LSR  $204F
1047-  4E 3F 3F   LSR  $3F3F
104A-  1F         ???
104B-  18         CLC
104C-  00         BRK
104D-  00         BRK
104E-  00         BRK
104F-  00         BRK
1050-  40         RTI
1051-  12         ???
1052-  FB         ???
1053-  75 35      ADC  $35,X
1055-  17         ???
1056-  07         ???
1057-  03         ???
1058-  01 40      ORA  ($40,X)
105A-  12         ???
105B-  31 15      AND  ($15),Y
105D-  00         BRK
105E-  00         BRK
105F-  00         BRK
1060-  00         BRK
1061-  00         BRK
1062-  00         BRK
1063-  00         BRK
1064-  AD 50 10   LDA  $1050
1067-  8D 59 10   STA  $1059
106A-  8D 5B 10   STA  $105B
106D-  AD 51 10   LDA  $1051
1070-  8D 5A 10   STA  $105A
1073-  8D 5C 10   STA  $105C
1076-  AE 52 10   LDX  $1052
1079-  A0 03      LDY  #$03
107B-  EE 5B 10   INC  $105B
107E-  AD 5B 10   LDA  $105B
1081-  D0 03      BNE  $1086
1083-  EE 5C 10   INC  $105C
1086-  88         DEY
1087-  C0 00      CPY  #$00
1089-  D0 F0      BNE  $107B
108B-  CA         DEX
```

branch to error message.

Array N$ found. Advance a further 7 places to length of N$(0). Place that address in $1050-1051.

Jump to find address for length and pointers of final item N$(500).

Error message.

Holds address of N$(500) length.

address of length & pointers for N$(0).

Gap between items being compared. Goes from 2510 to 1.

Address for lengths of two items being compared.

- swap flag

GET ADDRESSES for length of 1st. two items to be compared.

Using the gap found at $1052 (this moves to $1058), get the addresses for the lengths of the first two items to be compared; place in 1059-105A and 1058-105C.

SUPSORT is an assembly sort which sorts 500 or less records. It will sort 500 disordered records in about 30 seconds. (It will sort a reverse ordered list in about half that time.)

The program does the following:- 1. Finds the addresses for the lengths and pointers of an array called N$(501).
2. Then runs a 'shellsort' type sort. It compares items 251 apart (swaps if necessary), then items 117,53,232,7,3,1 apart on later runs through the list.
3. If only, say, 200 records are being sorted, it still runs through this sequence. As Applesoft sets all arrays to zero at the start, this does not matter.

By no means is this the ultimate sort for this type of sort. The times could be improved by at least a factor of two (I believe) if the 'Bubblesort' part at the end ran in two directions, and only checked the unsorted part of the array.

'SORTEM' is a program that calls 'SUPASORT' to sort its array. It dimensions an array N$(501), then loads records

SORTS

into this array from disc. It then calls 'SUPASORT' which sorts the records, and 'SORTEM' puts them back on disc. (It also displays the sorted records first, but this is of course unnecessary).

'C$RITE' takes ten records and writes them to disc 50 times; creating a text file of 500 disordered names.

'C$NANNUM' places 500 records onto disc, and 'RESORT' is an example of a program which tags the records as they are read off the disc.

'RESORT' loads records from the disc in reverse order, to see how the sort goes with a reverse order list. 'SSORT' is an attempt to use the 'SUPSORT' with 1000 records. Larger gaps are poked into the 'gap' part of the assembly program. 1000 records take about 3 minutes to sort.

C. Benson
Moorooka Qld

```
108C-  E0 00     CPX  #$00
108E-  D0 E9     BNE  $1079
1090-  A9 00     LDA  #$00
1092-  8D 60 10  STA  $1060
1095-  4C 9B 10  JMP  $109B
1098-  00        BRK
1099-  00        BRK
109A-  00        BRK
109B-  AD 5C 10  LDA  $105C
109E-  CD 4B 10  CMP  $104B
10A1-  D0 0B     BNE  $10AE
10A3-  AD 5B 10  LDA  $105B
10A6-  CD 4A 10  CMP  $104A
10A9-  D0 03     BNE  $10AE
10AB-  4C 90 11  JMP  $1190
10AE-  A2 00     LDX  #$00
10B0-  BD 59 10  LDA  $1059,X
10B3-  9D D4 10  STA  $10D4,X
10B6-  9D E2 10  STA  $10E2,X
10B9-  9D 73 11  STA  $1173,X
10BC-  9D 7A 11  STA  $117A,X
10BF-  BD 5B 10  LDA  $105B,X
10C2-  9D DA 10  STA  $10DA,X
10C5-  9D E8 10  STA  $10E8,X
10C8-  9D 77 11  STA  $1177,X
10CB-  9D 7E 11  STA  $117E,X
10CE-  E8        INX
10CF-  E0 02     CPX  #$02
10D1-  D0 DD     BNE  $10B0
10D3-  AD 40 12  LDA  $1240
10D6-  8D 5D 10  STA  $105D
10D9-  AD 31 15  LDA  $1531
10DC-  8D 5E 10  STA  $105E
10DF-  A2 01     LDX  #$01
10E1-  BD 40 12  LDA  $1240,X
10E4-  9D 23 11  STA  $1123,X
10E7-  BD 31 15  LDA  $1531,X
10EA-  9D 26 11  STA  $1126,X
10ED-  E8        INX
10EE-  E0 03     CPX  #$03
10F0-  D0 EF     BNE  $10E1
10F2-  4C 17 11  JMP  $1117
10F5-  00        BRK
10F6-  00        BRK
10F7-  00        BRK
10F8-  00        BRK
10F9-  00        BRK
10FA-  00        BRK
10FB-  00        BRK
10FC-  00        BRK
10FD-  00        BRK
10FE-  00        BRK
10FF-  00        BRK
1100-  00        BRK
1101-  00        BRK
1102-  00        BRK
1103-  00        BRK
1104-  00        BRK
1105-  00        BRK
1106-  00        BRK
1107-  00        BRK
1108-  00        BRK
1109-  00        BRK
110A-  00        BRK
110B-  00        BRK
110C-  00        BRK
```

Place 0 in 1060. (Set swap flag to zero.)

Check to see if 2nd. address is same as for length of N$(500). If so, go to 1190 to check for "sort completed."

GET ITEM ADDRESSES

Go to addresses for lengths and pointers. Place lengths in 105D and 105E. Place addresses for items into "SORT" and "SWAP"

```
110D-   00          BRK
110E-   00          BRK
110F-   00          BRK
1110-   00          BRK
1111-   00          BRK
1112-   00          BRK
1113-   00          BRK
1114-   00          BRK
1115-   00          BRK
1116-   00          BRK
1117-   AD 5E 10    LDA   $105E
111A-   F0 22       BEQ   $113E
111C-   AD 5D 10    LDA   $105D
111F-   F0 4F       BEQ   $1170
1121-   A2 00       LDX   #$00
1123-   BD 00 00    LDA   $0000,X
1126-   DD 5D 92    CMP   $925D,X
1129-   F0 04       BEQ   $112F
112B-   10 43       BPL   $1170
112D-   30 0F       BMI   $113E
112F-   E8          INX
1130-   EC 5D 10    CPX   $105D
1133-   F0 09       BEQ   $113E
1135-   EC 5E 10    CPX   $105E
1138-   F0 36       BEQ   $1170
113A-   E0 00       CPX   #$00
113C-   D0 E5       BNE   $1123
113E-   A2 00       LDX   #$00
1140-   EE 59 10    INC   $1059
1143-   AD 59 10    LDA   $1059
1146-   D0 03       BNE   $114B
1148-   EE 5A 10    INC   $105A
114B-   EE 5B 10    INC   $105B
114E-   AD 5B 10    LDA   $105B
1151-   D0 03       BNE   $1156
1153-   EE 5C 10    INC   $105C
1156-   E8          INX
1157-   E0 03       CPX   #$03
1159-   D0 E5       BNE   $1140
115B-   4C 9B 10    JMP   $109B
115E-   00          BRK
115F-   00          BRK
1160-   00          BRK
1161-   00          BRK
1162-   00          BRK
1163-   00          BRK
1164-   00          BRK
1165-   00          BRK
1166-   00          BRK
1167-   00          BRK
1168-   00          BRK
1169-   00          BRK
116A-   00          BRK
116B-   00          BRK
116C-   00          BRK
116D-   00          BRK
116E-   00          BRK
116F-   00          BRK
1170-   A2 00       LDX   #$00
1172-   BD 40 12    LDA   $1240,X
1175-   48          PHA
1176-   BD 31 15    LDA   $1531,X
1179-   9D 40 12    STA   $1240,X
117C-   68          PLA
117D-   9D 31 15    STA   $1531,X
1180-   E8          INX
1181-   E0 03       CPX   #$03
1183-   D0 ED       BNE   $1172
1185-   A9 05       LDA   #$05
1187-   8D 60 10    STA   $1060
118A-   4C 3E 11    JMP   $113E
118D-   00          BRK
118E-   00          BRK
118F-   00          BRK
1190-   AD 62 10    LDA   $1062
1193-   D0 1D       BNE   $11B2
1195-   EE 77 10    INC   $1077
1198-   AD 77 10    LDA   $1077
119B-   C9 58       CMP   #$58
119D-   D0 05       BNE   $11A4
119F-   A9 05       LDA   #$05
11A1-   8D 62 10    STA   $1062
11A4-   4C 64 10    JMP   $1064
11A7-   00          BRK
11A8-   00          BRK
11A9-   00          BRK
11AA-   00          BRK
11AB-   00          BRK
11AC-   00          BRK
11AD-   00          BRK
11AE-   00          BRK
11AF-   00          BRK
11B0-   00          BRK
11B1-   00          BRK
11B2-   AD 60 10    LDA   $1060
11B5-   F0 03       BEQ   $11BA
11B7-   4C 64 10    JMP   $1064
11BA-   60          RTS
11BB-   00          BRK
11BC-   00          BRK
11BD-   00          BRK
```

SORT

If 2nd. length zero, advance to next two items.

If 1st. length zero, SWAP.

Advance through items one letter at a time. If 2nd. smaller, "SWAP".

If length 2nd. runs out first, SWAP.

Advance addresses of lengths 3 positions to get addresses for next two items.

SWAP

Exchange lengths and pointers of the two items.

after running through item

Gap of 1 ? ($1062 = #05).

(a) No. Increment $1077 to get smaller gap.

(b) Yes. Adjacent items being compared. Jump to 1182 to see if any swap occured.

Any swap ?

(a) No. Run through list again.

(b) Yes. Finish.

C$RITE

```
]
]
C$RITE     26/11/'83
]
]
5  PRINT "THIS PROGRAM PLACES A TOTAL OF 500 NAMES ON DISC;
   TEN NAMES IN STRICT DISORDER"
7  PRINT : PRINT
10  DIM A$(10,50)
15  PRINT "LOADING NAMES TO ARRAY"
20  FOR I = 1 TO 10
30  ON I GOTO 510,520,530,540,550,560,570,580,590,600
50  FOR J = 1 TO 50
60 A$(I,J) = B$
65  PRINT B$;"  ";
70  NEXT J
75  NEXT I
80  PRINT "      "
81 D$ =  CHR$ (4)
82  PRINT D$;"OPEN NAM1"
83  PRINT D$;"WRITE NAM1"
90  FOR J = 1 TO 50
100  FOR I = 1 TO 10
110  PRINT A$(I,J)
115  NEXT I
116  NEXT J
200  PRINT "FF"
217  PRINT D$;"CLOSE NAM1"
340  GOTO 690
510 B$ = "PESTERJOHN"
515  GOTO 50
520 B$ = "ALBERTSON"
525  GOTO 50
530 B$ = "JOHNSMITH"
535  GOTO 50
540 B$ = "JASONCLARK"
545  GOTO 50
550 B$ = "THORERICKS"
555  GOTO 50
560 B$ = "JONLOVEDAY"
568  GOTO 50
570 B$ = "MACMATINS"
575  GOTO 50
580 B$ = "LESLIECARE"
585  GOTO 50
590 B$ = "WORKLESS"
595  GOTO 50
600 B$ = "ALECKSON"
605  GOTO 50
690  END
```

SORTS

C$NANNUM

```
]
]
C$NAMNUM     26/11/'83
]
]
2  PRINT "THIS PROGRAM PLACES 500 NAMES ON DISC IN STRICT DISORDER WITH AN
     ATTACHED 'STRING NUMBER' WITH EACH ONE"
7  PRINT : PRINT
10  DIM A$(10,50)
15  PRINT "LOADING NAMES TO ARRAY"
20  FOR I = 1 TO 10
30  ON I GOTO 510,520,530,540,550,560,570,580,590,600
50  FOR J = 1 TO 50
60 A$(I,J) = B$
65  PRINT B$;"  ";
70  NEXT J
80  NEXT I
81 D$ =  CHR$ (4)
85 N$ = "NAMNUM"
88  PRINT "     "
90  PRINT D$;"OPEN";N$
95  PRINT D$;"WRITE";N$
97  PRINT 500
100  FOR J = 1 TO 50
105  FOR I = 1 TO 10
110  PRINT A$(I,J)
140  PRINT  STR$ (K)
150 K = K + 1
160  NEXT I
170  NEXT J
210  PRINT D$;"CLOSE";N$
340  GOTO 690
510 B$ = "PESTERJOHN"
515  GOTO 50
520 B$ = "ALBERTSON"
525  GOTO 50
530 B$ = "JOHNSMITH"
535  GOTO 50
540 B$ = "JASONCLARK"
545  GOTO 50
550 B$ = "THORERICKS"
555  GOTO 50
560 B$ = "JONLOVEDAY"
568  GOTO 50
570 B$ = "MACMATINS"
575  GOTO 50
580 B$ = "LESLIECARE"
585  GOTO 50
590 B$ = "WORKLESS"
595  GOTO 50
600 B$ = "ALECKSON"
605  GOTO 50
690  END
```

Resort

```
RESORT     26/11/'83
]
]
5  LOMEM: 4644
10  DIM N$(501),R$(501)
15  PRINT "THIS PROGRAM IS TO SORT NAMES WITH RELATED STRING DATA"
16  PRINT : PRINT : PRINT
20 D$ =  CHR$ (4)
25 N$ = "NAMNUM"
30  PRINT D$;"BLOAD SUPSORT"
40  PRINT D$;"OPEN";N$
50  PRINT D$;"READ";N$
60  INPUT A1
70 A1 = A1 - 1
80  FOR I = 0 TO A1
90  INPUT N$(I)
91  IF I > 9 THEN 94
92 S$ = "00" +  STR$ (I)
93  GOTO 98
94  IF I > 99 THEN 97
95 S$ = "0" +  STR$ (I)
96  GOTO 98
97 S$ =  STR$ (I)
98 N$(I) = N$(I) + S$
99  INPUT R$(I)
100  NEXT I
120  PRINT D$;"CLOSE";N$
122  PRINT "I  ";I
125  PRINT "SORTING"
130  CALL 4086
135  FOR K = 0 TO A1
170  PRINT N$(K);"  ";
180  NEXT K
182  PRINT : PRINT
185 D$ =  CHR$ (4)
200  PRINT D$;"OPEN";N$
202  PRINT D$;"WRITE";N$
205 A2 = A1 + 1
210  PRINT A2
220  FOR I = 0 TO A1
222 L =  LEN (N$(I))
223 L1 = L - 2
224 L2 = L - 3
225 L2$ =  MID$ (N$(I),L1,3)
226 N$(I) =  MID$ (N$(I),1,L2)
228 L2 =  VAL (L2$)
230  PRINT N$(I)
232  PRINT R$(L2)
235  NEXT
240  PRINT D$;"CLOSE";N$
245  PRINT "     "
250  END
```

Read names from disc and add 3 space code.

name ØØ1
name ØØ2
name Ø53
name 437

Take code off the end.

SHOOTOUT

Shootout is a game requiring fast reflexes and keen eyesight. You are the fastest gunslinger in the west, and have been challenged by the Mexican gunfighter, El Ppa (amazing what some people's names spelt backwards translate as). His face (he don't look real mean, but he's quick on the trigger) appears on the screen. After a short pause the word DRAW also appears, with a beep if the easy game has been selected, without it the hard game is indicated.

When this happens press any key to fire. If you were quick enough, you win that shootout. If not, well, you get another chance (you can have up to 10 chances) unless that was the last battle. Your scores and his are totalled and the winner is announced.

El Ppa can be slowed down by increasing the number in line 230 or sped up by decreasing it.

Tony Humfrey
Parkes NSW

```
0  ONERR  GOTO 1
10  HOME :GAMES = 0:SHOOT = 0:DEAD = 0
20  VTAB 1
30  FOR A = 1 TO 40: PRINT "_";: NEXT
40  VTAB 11: FOR A = 1 TO 40: PRINT "_";: NEXT
50  VTAB 3: HTAB 16: FLASH : PRINT "SHOOTOUT": NORMAL : VTAB 5: PRINT "
     YOU ARE THE FASTEST GUNSLINGER IN THE WEST(OR EAST,FOR THAT MATTER)A
    ND YOU    HAVE BEEN CHALLENGED BY THE MEXICAN GUN-FIGHTER EL PPA.";
60  PRINT "YOU HAVE TO BEAT HIM IN AGUNFIGHT OR LOSE YOR TITLE AS THE BES
    T  GUNSLINGER!": VTAB 15: HTAB 14: INVERSE : PRINT "INSTRUCTIONS": NORMAL
    : PRINT : PRINT "   WHEN EL PPA DRAWS HIS GUN,YOU HAVE 1 SECOND IN W
    HICH TO DRAW   YOUR OWN GUN BY PRESSING ANY KEY"
70  PRINT "IF YOU BEAT HIM MORE THAN HALF THE TIMESYOU PLAY HIM,YOU WILL
    BE HERALDED AS THEWINNER": PRINT : INPUT "HOW MANY SHOOTOUTS(UP TO T
    EN)->";SHOOT: IF SHOOT > 10 THEN 70
71  HOME : INPUT "HARD GAME (Y/N)";D$
80  REM  GEANHD
90 C =  INT ( RND (1) * 10) + 100
100  HOME : INVERSE : VTAB 8: HTAB 18: PRINT "  ": VTAB 8: HTAB 21: PRINT
    "  "
110  VTAB 10: HTAB 20: PRINT " ": VTAB 11: HTAB 19: PRINT ". ."
120  VTAB 12: HTAB 17: PRINT " ": VTAB 12: HTAB 23: PRINT " "
130  VTAB 13: HTAB 18: PRINT "   "
140  VTAB 6: NORMAL : HTAB 16: PRINT "_________": FOR V = 7 TO 15: VTAB
    : HTAB 15: PRINT "I": VTAB V: HTAB 25: PRINT "I": NEXT V: VTAB 15: HTAB
    16: PRINT "_________"
150  FOR A = 1 TO C
160  IF  PEEK ( - 16384) > 127 THEN 310
170  NEXT A
180 V =  INT ( RND (1) * 20) + 1
190 H =  INT ( RND (1) * 36) + 1
200  IF V >  = 6 AND V <  = 16 THEN 180
210  VTAB V: HTAB H
215  IF  LEFT$ (D$,1) = "Y" THEN  INVERSE : PRINT "DRAW": NORMAL : GOTO 2
    30
220  INVERSE : PRINT "DRAW": NORMAL : REM INSERT CTRL-G INTO "DRAW"
230  FOR B = 1 TO 10
240  IF  PEEK ( - 16384) > 127 THEN 270
250  NEXT B
260  HOME : PRINT "BANG ! YOU'RE DEAD":DEAD = DEAD + 1: GET A$: GET A$: GOTO
    290
270  FOR T = 1 TO 1500: NEXT T: HOME : PRINT "YOU GOT HIM!!":GAMES = GAME
    S + 1
280  GET F$: GET F$
290  FOR M = 1 TO 1500: NEXT : IF DEAD + GAMES =  > SHOOT THEN 330
300  GOTO 80
310  FOR S = 1 TO 100:D =  PEEK ( - 16336): NEXT S
320  HOME : VTAB 12: INVERSE : PRINT "YOU TRIED TO CHEAT,BUT YOU DIDN'T W
    IN.": NORMAL : GET A$: GET S$: GOTO 80
330  HOME : PRINT "HE WON ";DEAD;" GAMES:YOU WON ";GAMES;" GAMES": FOR A =
    1 TO 1500: NEXT
340  IF DEAD > GAMES THEN 380
350  IF DEAD = GAMES THEN 400
360  IF DEAD < GAMES THEN 410
370  END
380  HOME : VTAB 9: HTAB 17: PRINT "EL PPA": VTAB 10: HTAB 17: PRINT "L
       P": VTAB 11: HTAB 17: PRINT "P    P": VTAB 12: HTAB 17: PRINT "P
       L": VTAB 13: HTAB 17: PRINT "APP LE": VTAB 2: PRINT "WHO DO WE SUP
    PORT-WE SUPPORT THE ONLY-"
390  GOTO 420
400  HOME : PRINT "IT IS A TIE!WE DEMAND A REMATCH!": FOR A = 1 TO 1500: NEXT
    : GOTO 10
410  HOME : PRINT "_____________________________________": PRINT : PRINT
    "    WE KNEW YOU COULD DO IT FOR US":
420  FOR A = 1 TO 3000: NEXT : HOME : PRINT "ANOTHER GAME,PODNER(Y/N)->";
    : INPUT Y$: IF  LEFT$ (Y$,1) = "Y" THEN  GOTO 10
430  END
```

WORMS

'WORMS' is a game where you, as a worm, must destroy your enemy by totally blocking him so that he is forced to hit either one of your segments, his segments, the obstacles or the border. Your 'worm' starts off in a random position on the right hand side of the screen and there is a short delay before the action starts – this is so that you can pick up where you are. Once the game starts you have to complete ten rounds to win.

The game incorporates a feature that allows you to change the controlling keys to suit your preference. The only key not allowed to be used in this way is the right arrow key.

Before you can start this game you must first create the text file that' "Worms' uses. It is called 'High Scores'. To do this, type in the text file creator and run it. The disk will whirl a few seconds and then stop. Now, type in the 'Worms' program and save. The program is now ready to run.

I made the game on a black and white monitor and so I used colours that suited it. However, if you want to change the colour of the border and obstacles, the command is on line 50. The colour of the computer worm is controlled by line 73 and your worm's colour is controlled by line 225. You might want to change line 55 as well, but DON'T change the COLOR = 0 on that line.

Michael Lee
Torrens ACT

```
TEXT FILE CREATOR
b/ Michael Lee
10 D$ = CHR$ (4)
20 PRINT D$"OPEN HIGH SCORES"
30 PRINT D$"WRITE HIGH SCORES"
40 FOR Q = 1 TO 20
50 PRINT "0": PRINT "-------"
60 NEXT Q
70 PRINT D$"CLOSE HIGH SCORES"
80 NEW
The "0" on line 50 is meant to be a ZERO.
If this is not there the WORMS program will respond with an
error.
```

```
1  TEXT : HOME : SPEED= 255
2  REM  INIT THE VARIABLES
3  REM
5 U$ = "T":D$ = "G":R$ = "H":L$ = "F": DIM C(21): DIM C$(21)
7  REM
8  REM  GOTO INSTRUCTIONS
9  REM
10  GOSUB 2060
20  GOTO 2030
40 X = 1:Y =  INT ( RND (1) * 37 + 1):X1 = 38:Y1 =  INT ( RND (1) * 37 + 1)
41 A = 4:Z = 3
45  VTAB 21: PRINT "YOU ARE ON LEVEL "G
50  GR : COLOR= 15
51  FOR G = 1 TO G
52 X2 =  INT ( RND (1) * 26) + 7:Y2 =  INT ( RND (1) * 40):Z2 =  INT ( RND (1)
 * 40):A2 =  INT ( RND (1) * 26) + 7:B2 =  INT ( RND (1) * 26) + 7:C2 =  INT (
 RND (1) * 40): IF (Z2 < 2) OR (Z2 > 37) OR (Y2 < 2) OR (Y2 > 37) THEN  GOTO 5
2
53  VLIN Y2,Z2 AT X2
54  HLIN A2,B2 AT C2: NEXT
55  VLIN 0,39 AT 0: VLIN 0,39 AT 39: HLIN 0,39 AT 0: HLIN 0,39 AT 39: COLOR= 5
: PLOT X + 1,Y: COLOR= 2: PLOT X1 - 1,Y1: FOR Q = 1 TO 1000: NEXT Q: COLOR= 0:
 PLOT X + 1,Y: PLOT X1 - 1,Y1
56  GOTO 70
60 A =  INT ( RND (1) * 4 + 1)
70 T =  INT ( RND (1) * 100 + 1): IF T < G THEN  GOTO 60
73  COLOR= 5
75  IF A = 1 THEN Y = Y - 1
80  IF A = 2 THEN Y = Y + 1
85 M =  PEEK ( - 16336)
90  IF A = 3 THEN X = X - 1
95 M =  PEEK ( - 16336)
100  IF A = 4 THEN X = X + 1
110  IF  SCRN( X,Y) <  > 0 THEN  GOTO 1000
120  PLOT X,Y
122  REM
125  REM   GET THE DIRECTION KEYS (HUMAN)
127  REM
130  IF  PEEK ( - 16384) > 127 THEN  GET A$
140  IF (A$ = U$) AND (Z <  > 2) THEN Z = 1
150  IF (A$ = D$) AND (Z <  > 1) THEN Z = 2
160  IF (A$ = L$) AND (Z <  > 4) THEN Z = 3
170  IF (A$ = R$) AND (Z <  > 3) THEN Z = 4
175 S = S + 10 + G: VTAB 22: PRINT "SCORE="S"  "
180  IF Z = 1 THEN Y1 = Y1 - 1
190  IF Z = 2 THEN Y1 = Y1 + 1
200  IF Z = 3 THEN X1 = X1 - 1
210  IF Z = 4 THEN X1 = X1 + 1
220  IF  SCRN( X1,Y1) <  > 0 THEN :P = 1: GOTO 2000
225  COLOR= 2
230  PLOT X1,Y1
240 M =  PEEK ( - 16336)
990  GOTO 70
1000  IF A = 1 THEN Y = Y + 1
1001  IF A = 2 THEN Y = Y - 1
1002  IF A = 3 THEN X = X + 1
1005  IF A = 4 THEN X = X - 1
1010  IF (A = 1) OR (A = 2) THEN  GOTO 1030
1020  IF (A = 3) OR (A = 4) THEN  GOTO 1500
1023  REM
1025  REM  CHECK TO SEE IF THE COMPUTER HAS HIT A WALL AND TO TURN THE WORM IF
 NEED BE
1027  REM
1030 B =  SCRN( X - 1,Y):C =  SCRN( X + 1,Y)
1040  IF (B = 0) AND (C = 0) THEN A =  INT ( RND (1) * 2 + 3): GOTO 70
1050  IF B = 0 THEN A = 3: GOTO 70
1060  IF C = 0 THEN A = 4: GOTO 70
1070  GOTO 2000
1500 B =  SCRN( X,Y - 1):C =  SCRN( X,Y + 1)
1510  IF (B = 0) AND (C = 0) THEN A =  INT ( RND (1) * 2 + 1)
1520  IF B = 0 THEN A = 1: GOTO 70
1530  IF C = 0 THEN A = 2: GOTO 70
1540  GOTO 2000
1800  REM
1900  REM  IF ANY WORM GETS HIT THE PROGRAM COMES HERE.
1905  REM
2000 M =  - 16336:M =  PEEK (M) +  PEEK (M) +  PEEK (M) +  PEEK (M) +  PEEK (M
) +  PEEK (M) +  PEEK (M) +  PEEK (M) +  PEEK (M) +  PEEK (M) +  P
EEK (M) +  PEEK (M): FOR Q = 1 TO 2500: NEXT
2010  TEXT : HOME
2020  IF P = 1 THEN  GOTO 2050
2022  REM
2023  REM  IF YOU HAVE BEATEN THE COMPUTER,COME HERE.
2024  REM
2025              IF G = 10 THEN  INVERSE : PRINT "CONGRATULATIONS": PRINT : P
RINT "YOU HAVE SUCESSFULLY DRIVEN AWAY ALL     PREDITORS.YOUR SCORE IS "S: NORM
AL : FOR Q = 1 TO 5000: NEXT : GOTO 3000
2029  INVERSE : PRINT "YOU HAVE WON": NORMAL
2030 G = G + 1: PRINT : PRINT "YOU ARE NOW ON LEVEL "G: PRINT "WATCH OUT I'M G
OING TO ATTACK!!!"
```

```
2032  REM
2035  REM  IF YOU WON BUT NOT DONE TEN ROUNDS THEN REPEAT FROM HERE.
2037  REM
2040  FOR Q = 1 TO 2500: NEXT : HOME : GR : COLOR= 15: GOTO 40
2042  REM
2045  REM  IF YOU LOSE YOUR GROUNDS COME HERE
2047  REM
2050  PRINT "YOU HAVE LOST YOUR GROUNDS ON LEVEL "G: PRINT "YOUR SCORE IS "S:
FOR Q = 1 TO 1000: NEXT Q
2051 W = 1: GOTO 3000
2052  REM  GOTO 3000 TO HALL OF FAME.
2053  POKE  - 16368,0
2055  PRINT : INPUT "DO YOU WANT TO PLAY AGAIN?"¡A$: IF A$ = "Y" THEN S = 0:G
= 0:H1 = 0:P = 0: GOSUB 2230: GOTO 2030
2056  PRINT "GOODBYE AND THANKS FOR THE LAND!!!": END
2058  REM
2060  REM  INTRODUCTION
2065  REM
2070  HOME : VTAB 9: HTAB 18: INVERSE : PRINT "WORMS": NORMAL
2080  HTAB 15: PRINT "PROGRAMED BY"
2090  HTAB 15: PRINT "MICHAEL--LEE"
2100  PRINT : HTAB 15: PRINT "19/6/83"
2120  FOR Q = 1 TO 1000: NEXT
2140  HOME : INVERSE : PRINT "                    WORMS                    "
2145  NORMAL
2150  PRINT : PRINT "YOU ARE A WORM FIGHTING FOR POSSESION OFA PIECE OF FERTIL
E LAND"
2160  PRINT : PRINT "YOUR ENEMIES WILL TRY TO RUN YOU OUT OF IT AND YOU MUST S
TOP THEM"
2170  PRINT : PRINT "THE LONGER YOU STAY ALIVE THE MORE       POINTS YOU GET,HO
WEVER TO SUCCESSFULLY  WIN THE GAME YOU MUST FIGHT THE ENEMY    TEN TIMES"
2180  PRINT : PRINT "AS YOU PROGRESS THE ENEMY WILL GET MORE INTELLIGENT AND T
HE TERRAIN MORE         CHALLENGING"
2190  PRINT : FLASH : PRINT "GOOD LUCK": NORMAL
2200  VTAB 22: PRINT "<<<<<<<<PRESS-ANY-KEY-TO-CONTINUE>>>>>>>>"
2210  IF  PEEK ( - 16384) < 128 THEN  GOTO 2210
2220  POKE  - 16368,0
2230  HOME : INVERSE : PRINT "THESE ARE YOUR COMMANDS"
2240  NORMAL : PRINT : PRINT "  "U$"=UP"
2250  PRINT : PRINT "  "D$"=DOWN"
2260  PRINT : PRINT "  "L$"=LEFT"
2270  PRINT : PRINT "  "R$"=RIGHT"
2280  PRINT : PRINT : INPUT "DO YOU WANT TO CHANGE THESE COMMANDS?"¡Q$: VTAB 1
2: PRINT "                                                  "
2282  IF  LEFT$ (Q$,1) = "Y" THEN  VTAB 3: HTAB 3: GET U$: PRINT U$: VTAB 5: H
TAB 3: GET D$: PRINT D$: VTAB 7: HTAB 3: GET L$: PRINT L$: VTAB 9: HTAB 3: GET
 R$: PRINT R$: VTAB 12: INPUT "ARE THESE ALRIGHT?"¡Q$: IF  LEFT$ (Q$,1) = "N"
THEN Q$ = "Y": GOTO 2282
2300  VTAB 12: PRINT "<<<<<<<<<<<<<<<PRESS ANY KEY>>>>>>>>>>>>>>>": VTAB 12: GET
Q$: PRINT Q$: HOME : RETURN
2310  REM
2320  REM  OPEN THE HIGH SCORE FILE AND READ IT.
2330  REM
3000  IF W <  > 1 THEN S = S + 1000
3010 A$ =  CHR$ (4)
3020  PRINT A$"OPEN HIGH SCORES"
3030  PRINT A$"READ HIGH SCORES"
3040  FOR Q = 1 TO 20
3050  INPUT C(Q)
3060  INPUT C$(Q)
3065  NEXT
3070  PRINT A$"CLOSE HIGH SCORES":Z = 0
3071  REM  CHECK TO SEE IF SCORE IS BEATEN
3072  FOR Q = 1 TO 20: IF Z = 1 THEN  GOTO 3075
3073  IF S >  = C(Q) THEN Z = 1:ML = Q: FOR W = 20 TO Q STEP  - 1:C(W + 1) = C
(W):C$(W + 1) = C$(W): NEXT W
3075  NEXT Q
3076  REM
3077  REM  IF YOU BEAT A HIGH SCORE THEN TYPE IN AND SAVE NAME.
3078  REM
3080  HOME : IF Z = 1 THEN  PRINT "YOU HAVE BEATEN A HIGH SCORE PLEASE TYPEIN
YOUR NAME----------------------------": INPUT C$(ML):C(ML) = S
3100  REM
3140  REM  SORT NAMES AND PUSH THEM ALL DOWN IF SCORE IS BEATEN.ACTUALLY 3073
IS THE REAL SORT.
3150  REM
3160  PRINT A$"OPEN HIGH SCORES"
3170  PRINT A$"WRITE HIGH SCORES"
3180  FOR Q = 1 TO 20
3190  PRINT C(Q)
3200  PRINT C$(Q)
3210  NEXT
3220  PRINT A$"CLOSE HIGH SCORES"
3230 W = 0:J = 0: HOME
3240  PRINT "<<<<<<<<<<H-A-L-L--OF--F-A-M-E>>>>>>>>>>"
3250  VTAB 2: PRINT " NAMES                    SCORES"
3260  FOR Q = 1 TO 20: PRINT Q")"C$(Q)
3270  VTAB Q + 2: HTAB 27: PRINT C(Q)
3280  NEXT
3290  POKE  - 16368,0: PRINT "################PRESS ANY KEY################"
3300  IF  PEEK ( - 16384) < 128 THEN  GOTO 3300
3310  GOTO 2053
```

APPLE II

GRAPHICS DRAWER

Graphics Drawer enables the user to draw graphics on Hi-Res Page 2 using the Apple's keyboard.

The controls are as follows:

I – draw line upwards
J – draw line left
K – draw line right
M – draw line downwards
O – draw line diagonally up – right
U – draw line diagonally down – left
, – draw line diagonally down · right (NB all the above keys are for movement.)

C – colour (0&4 – black, 1 – green, 2 – violet, 3&7 – white, 5 – orange, 6 – green)

D – Increment – how many dots plotted per keypress.

An example of Graphics Drawer has been included to show its capabilities. There is also another program – Sample Pattern Routines which has some interesting routines.

Tony Humfrey
Parkes NSW

```
10 X = 139:Y = 90: HGR2 :COL = 3:INC = 10
11 X1 = X:Y1 = Y
20 GET M$
30 IF M$ = "I" THEN Y = Y - INC: IF Y = < 0 THEN Y = 0
35 IF M$ = "O" THEN Y = Y - INC:X = X + INC: IF Y = < 0 THEN Y = 0: IF
    X = > 279 THEN X = 279
40 IF M$ = "M" THEN Y = Y + INC: IF Y = > 191 THEN Y = 191
45 IF M$ = "," THEN Y = Y + INC:X = X + INC: IF Y = > 191 THEN Y = 191:
    IF X = < 0 THEN X = 0
50 IF M$ = "K" THEN X = X + INC: IF X = > 279 THEN X = 279
55 IF M$ = "N" THEN X = X - INC:Y = Y + INC: IF X = > 279 THEN X = 279:
    IF Y = > 191 THEN Y = 191
60 IF M$ = "J" THEN X = X - INC: IF X = < 0 THEN X = 0
65 IF M$ = "U" THEN X = X - INC:Y = Y - INC: IF X = < 0 THEN X = 0: IF
    Y = < 0 THEN Y = 0
70 IF M$ = "D" THEN 2000
75 IF M$ = "E" THEN  PRINT "": END : REM  [ CTRL-G ]
80 IF M$ = "C" THEN 1000
81 HCOLOR= COL: HPLOT X1,Y1 TO X,Y
82 GOTO 11
1000  POKE  - 16300,0: POKE  - 16303,0: HOME : VTAB 12: PRINT "COLOUR=>":
     GET COL: POKE  - 16299,0: POKE  - 16304,0: GOTO 11
2000  POKE  - 16300,0: POKE  - 16303,0: HOME : VTAB 12: PRINT "INCREMENT=
     >": GET INC: POKE  - 16299,0: POKE  - 16304,0: GOTO 11
```

Sample Pattern Routines for Graphics Drawer

```
1  HOME : PRINT "PRESS A NUMBER TO RUN PROGRAMS 1 THRU 0": GET A
2  ON A GOTO 10,100,200,300,400,500,600,700,800,900
5  HOME
8 0
10 X =  INT ( RND (1) * 38) + 1
15 Y =  INT ( RND (1) * 23) + 1
20  HTAB X: VTAB Y: PRINT "*"
30  GOTO 10
100  FOR A = 0 TO 255
110  PRINT  CHR$ (A);
120  NEXT A
125  GET A$: GOTO 1
200  HGR2
205  HCOLOR= 3
210 Y =  INT ( RND (1) * 190) + 1
220 X =  INT ( RND (1) * 255) + 1
230  HPLOT X,Y
240  GOTO 210
300  HGR2
305  HCOLOR= 3
310 Y =  INT ( RND (1) * 190) + 1
320 X =  INT ( RND (1) * 254) + 1
330 B =  INT ( RND (1) * 190) + 1
340 A =  INT ( RND (1) * 254) + 1
350  HPLOT X,Y
355  HPLOT  TO A,B
360  GOTO 310
400  HGR
410  FOR A = 1 TO 7
415  IF A = 4 OR A = 4 THEN  GOTO 410
420  HCOLOR= A
425  FOR B = 1 TO 279
430  HPLOT B,0 TO B,160
440  NEXT B
450  HGR : NEXT A
460  GOTO 1
500  HGR2
510  FOR A = 1 TO 7
515  IF A = 4 THEN  GOTO 510
520  HCOLOR= A
525  FOR B = 0 TO 191
530  HPLOT 0,B TO 279,B
540  NEXT B
550  HGR2 : NEXT A
560  GOTO 1
600  HGR2 :H =  INT ( RND (1) * 7) + 1
605  IF H = 4 THEN 600
610  HCOLOR= H
615  FOR B = 0 TO 191
620  HPLOT 0,B: HPLOT  TO B,0
625  NEXT B
630  GOTO 600
700  HGR2
710 X =  INT ( RND (1) * 279) + 1
720  FOR Y = 0 TO 191
721  HPLOT X,Y
725  NEXT Y
730  GOTO 710
800  HGR2 :H =  INT ( RND (1) * 7) + 1
805  IF H = 4 THEN 800
809 X = 0:Y = 0
810  HCOLOR= H
815  HPLOT X,Y: HPLOT 0,Y TO X,0
820 X = X + 1:Y = Y + 1: IF X > 279 THEN X = 279: IF Y > 191 THEN Y = 191
     : GOTO 810
```

COPY PROTECTOR

This program prevents copying, and, in fact, looking at programs on your disk. It uses the RWTS subroutine to change the directory file location. It is left to you to decide how to encompass this into your own greeting program. The basic idea of this program makes it quite flexible and it can be easily expanded as I will describe later.

Bytes Accessed:
$303 – Volume Number
$304 – Track Number
$305 – Sector Number
$306 – Command (01-Read 02-Write)

If you want to look at any sector on your disk or in fact when you set up your copy protector system, you just change the above four locations as required and type 315G. The sector read or written will be from $2000 – $20FF. This buffer can be changed by altering locations $308(low-byte) and $309 (high-byte).

Take a newly initialised disk and type in the program below and save it as the greeting program.

```
10 HOME
20 ?CHR$(4);"BRUN DC" 30
?CHR$(4);"CATALOG"
```

CALL-151 and type in the hex DC program as given and bsave it as the file name in 20.

Using the 315G procedure above, copy the directory of track $11-sector $OF into track $22-sector $OF. You now have a real directory in track $11 and a false one in track $22-sector $OF. When the disk is booted the program will change the VTOC so that a catalog will show the false directory. In fact, DOS can not load a program unless it is contained in the directory. To get the real directory back you simply CALL-151 then 333G.

The system I use is slightly different to the above and was first placed on a half full disk. The difference is that the greeting program is different for each directory but has the same file name.

The false directory is exactly the same as the above but the real directory points to a different track/sector list.

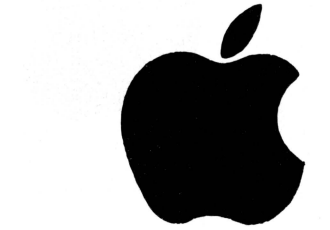
The easiest way to accomplish this is to save the real directories hello program as normal. Then save the false directories hello program on another disk or under a new file name on the disk you are copy protecting. Now transfer the false directory containing the files, HELLO and DC, into track $22-sector $OF by the "315G" method, using track $22-sector $OE as the track/sector list for the hello program. Write the track/sector list into the data buffer, using track $22-sector $OD as the first and only file location, and save this into track $22-sector $OE. Then load the actual tokenised sector (NB – the above hello program occupies only one sector), from the disk used to save the false hello program, into the data buffer so that it can be saved into track $22-sector $OD.

When the disk is booted the false directory will be used and the catalog will show the two files, HELLO and DC. This allows you to still retrieve the real directory even if the disk has not been booted.

Also, it is important to adjust the track-bit maps to show the sectors you have used with the RWTS. All the relative information can be gained in the DOS manual under storage of files. For the system to work, both hello programs must first BRUN DC or the false directory must first BRUN DC and the VTOC must point to the false directory while you are not using the disk. This is done by BRUNing DC for the first and second cases or booting the disk in the first case.

Michael Werner
(Send us your address, Michael!)

```
--- HEX DUMP ---

02EA: A9 01 8D 03 03 8D 0C 03 A9 11
      8D 04 03 A9 00 8D 05 03 20 1D
      03 60 01 60 01 01 01 11 0F 11
      03 00 20 00 00 01 00 01 60 01
      00 01 EF D8 A9 03 A0 00 20 D9
      03 60 20 58 FC 20 15 03 A0 01
      A9 22 99 00 20 A9 02 8D 0C 03
      20 15 03 60 A0 11 8C 04 03 A0
      00 8C 05 03 A0 01 8C 0C 03 20
      15 03 A0 01 A9 11 99 00 20 A9
      02 8D 0C 03 20 15 03 60
```



APPLESOFT COMMAND

With this short routine, you can type BASIC commands using a single key with the control key. The keys and keywords I have chosen are shown in the table at the end of the program.

BASIC commands begin at $D0D0 and occupy consecutive locations to $D25E. The first seven keys (@ to F) access commands on page $D0. The keys G to Y (excluding H, M and U) access keywords on page $D1, whilst Z accesses a keyword on page $D2.

The number of keys which access pages $D0, $D1, and $D2 could be changed altering the numbers in locations $0333 and $033D respectively.

The keywords could be changed by substituting the least significant byte of the address of the new command for one of those on the list.

Enter the monitor and type in the program beginning at $0300. Save the program by typing:

BSAVE ACE, A$300,L$91

To run the program type BRUN ACE from disk or BLOAD ACE followed by CALL 768. If you begin the program from the monitor with 3006, you must re-enter Applesoft by typing 3D0G, as typing Control-C will produce CALL. Before running a program in Applesoft hit the reset button to revert to the normal input routine.

The program works by passing all input through ACE. If

```
]CALL-151

*300.390

0300- A9 D2 85 1B A9 03 85 1D
0308- 85 39 A9 76 85 1C A9 00
0310- 85 19 A9 19 85 38 4C EA
0318- 03 20 4A FF A5 19 D0 2E
0320- A5 45 20 1B FD 85 45 C9
0328- 9B 90 04 20 3F FF 60 29
0330- 7F A8 C9 07 B0 06 C6 1B
0338- C6 1B D0 06 C9 1A B0 02
0340- C6 1B B1 1C F0 20 85 06
0348- E6 19 A9 00 85 07 A5 45
0350- A4 07 A5 06 85 1A B1 1A
0358- C9 80 B0 08 09 80 E6 07
0360- 85 45 D0 C7 85 45 A9 00
0368- 85 19 A9 D2 C5 1B F0 04
0370- E6 1B D0 F8 F0 B5 EF D3
0378- D6 F9 DA E9 DE 93 00 9A
0380- 56 4F 90 00 49 29 17 10
0388- 09 A4 EF 00 64 25 C7 A9
0390- 3B
```

```
0800                    1           TTL "APPLESOFT COMMAND ENTRY (ACE)"
0800                    2   * BY JOHN GALLAGHER. FEB.83
0300                    3           ORG $300
0300                    4           OBJ $800
0006                    5   ASTR    EPZ $06             ;TEMP.STORE A REG.
0007                    6   CHRCNT  EPZ $07             ;KEYWORD CHAR.COUNT
001A                    7   WRDADD  EPZ $1A             ;STORE KEYWORD ADDRESS
001C                    8   TBLADD  EPZ $1C             ;STORE LOOKUP TBL ADDRESS
0019                    9   WRDEND  EPZ $19             ;KEYWORD END FLAG
0038                   10   INHOOK  EPZ $38             ;INPUT HOOK
0045                   11   ASAVE   EPZ $45             ;A REG.STORE
0319                   12   START   EQU $319            ;INPUT ROUTINE
032B                   13   RETURN  EQU $32B
032F                   14   NXTWRD  EQU $32F            ;NEXT KEYWORD
034E                   15   NXTCHR  EQU $34E            ;GET NEXT CHAR.
0366                   16   NXTKEY  EQU $366            ;PREPARE FOR NEXT KEY
0376                   17   LKTBL   EQU $376            ;LOOKUP TABLE BEGINS
03EA                   18   EXIT    EQU $3EA            ;EXIT THRU I/O UPDATE
FD1B                   19   KEYIN   EQU $FD1B           ;READ KEYBOARD
FF4A                   20   IOSAVE  EQU $FF4A           ;SAVE REGISTERS
FF3F                   21   IOREST  EQU $FF3F           ;RESTORE REGISTERS
0300                   22   *
0300                   23   * INITIALIZATION
0300                   24   *
0300 A9 D2             25           LDA #$D2            ;HIGH PAGE KEYWORD ADDRESS
0302 85 1B             26           STA WRDADD+1
0304 A9 03             27           LDA /START
0306 85 1D             28           STA TBLADD+1
0308 85 39             29           STA INHOOK+1
030A A9 76             30           LDA #LKTBL
030C 85 1C             31           STA TBLADD
030E A9 00             32           LDA #$00
0310 85 19             33           STA WRDEND          ;CLEAR WORD END FLAG
0312 A9 19             34           LDA #START
0314 85 38             35           STA INHOOK
0316 4C EA 03          36           JMP EXIT            ;EXIT THRU I/O UPDATE
0319                   37   * START INPUT ROUTINE
0319 20 4A FF          38           JSR IOSAVE
031C A5 19             39           LDA WRDEND          ;IF NOT END OF WORD
031E D0 2E             40           BNE NXTCHR          ;GET NEXT CHARACTER
0320 A5 45             41           LDA ASAVE
0322 20 1B FD          42           JSR KEYIN
0325 85 45             43           STA ASAVE
0327 C9 9B             44           CMP #$9B            ;CHECK FOR CTRL.KEY
0329 90 04             45           BCC NXTWRD          ;IF CTRL GET KEYWORD
032B                   46   * RETURN
032B 20 3F FF          47           JSR IOREST          ;RESTORE & RETURN
032E 60                48           RTS
032F                   49   * NXTWRD
032F 29 7F             50           AND #$7F            ;REMOVE MSB
0331 A8                51           TAY
0332 C9 07             52           CMP #$07            ;IF NOT @-F CONTINUE
0334 B0 06             53           BCS ]1
0336 C6 1B             54           DEC WRDADD+1        ;IF @-F THEN
0338 C6 1B             55           DEC WRDADD+1        ;DEC TO $D0
033A D0 06             56           BNE ]2
033C C9 1A             57   ]1      CMP #$1A            ;IF Z LEAVE AT $D2
033E B0 02             58           BCS ]2
0340 C6 1B             59           DEC WRDADD+1        ;IF G-Y DEC TO $D1
0342 B1 1C             60   ]2      LDA (TBLADD),Y      ;LOOKUP INDEX
0344 F0 20             61           BEQ NXTKEY          ;IF KEY NOT USED RETURN
```

ENTRY

CTRL is pressed, the key following it is used to generate an index to obtain the least significant byte of the address of the BASIC command which is stored in a table beginning at $0376. This byte is then stored in $1A.

The most significant byte is stored in $1B and has an initial value of $D2. This is decremented to $D0 if keys @ to F are pressed and to $D1, if any other key apart from Z is pressed.

BASIC commands are stored with the MSB set only for the last character, and this is used to clear the word end flag ($19). A character counter ($07) provides the index to obtain each character from the keyword after its address has been located.

The initialisation routine sets the input hook to the beginning of the ACE input routine at $0319.

J. Gallagher
Paradise Park

```
0346 85 06      62          STA ASTR            ;STORE LO BYTE KEYWORD ADDRESS
0348 E6 19      63          INC WRDEND          ;SET WORD END FLAG
034A A9 00      64          LDA #$00
034C 85 07      65          STA CHRCNT          ;CLEAR CHR COUNT
034E            66  * NXTCHR
034E A5 45      67          LDA ASAVE
0350 A4 07      68          LDY CHRCNT
0352 A5 06      69          LDA ASTR            ;GET LO BYTE KEYWORD ADDRESS
0354 85 1A      70          STA WRDADD    .
0356 B1 1A      71          LDA (WRDADD),Y      ;GET NEXT CHARACTER
0358 C9 80      72          CMP #$80            ;LAST CHARACTER ?
035A B0 08      73          BCS >3
035C 09 80      74          ORA #$80
035E E6 07      75          INC CHRCNT
0360 85 45      76          STA ASAVE
0362 D0 C7      77          BNE RETURN
0364 85 45      78  ^3      STA ASAVE
0366            79  * NXTKEY
0366 A9 00      80          LDA #$00
0368 85 19      81          STA WRDEND          ;CLEAR WORD END FLAG
036A A9 D2      82          LDA #$D2
036C C5 1B      83  ^5      CMP WRDADD+1
036E F0 04      84          BEQ >4
0370 E6 1B      85          INC WRDADD+1        ;INC TO $D2
0372 D0 F8      86          BNE <5
0374 F0 B5      87  ^4      BEQ RETURN
0376            88  * LOOK UP TABLE
0376 EF         89          HEX EF              ;@=TEXT
0377 D3         90          HEX D3              ;A=FOR
0378 D6         91          HEX D6              ;B=NEXT
0379 F9         92          HEX F9              ;C=CALL
037A DA         93          HEX DA              ;D=DATA
037B E9         94          HEX E9              ;E=READ
037C DE         95          HEX DE              ;F=INPUT
037D 93         96          HEX 93              ;G=GO TO
037E 00         97          HEX 00              ;H NOT USED
037F 9A         98          HEX 9A              ;I=IF
0380 56         99          HEX 56              ;J=FLASH
0381 4F        100          HEX 4F              ;K=INVERSE
0382 90        101          HEX 90              ;L=LET
0383 00        102          HEX 00              ;M NOT USED
0384 49        103          HEX 49              ;N=NORMAL
0385 29        104          HEX 29              ;O=HOME
0386 17        105          HEX 17              ;P=HPLOT
0387 10        106          HEX 10              ;Q=HCOLOR=
0388 09        107          HEX 09              ;R=HGR2
0389 A4        108          HEX A4              ;S=GO SUB
038A EF        109          HEX EF              ;T=THEN
038B 00        110          HEX 00              ;U NOT USED
038C 64        111          HEX 64              ;V=VTAB
038D 25        112          HEX 25              ;W=HTAB
038E C7        113          HEX C7              ;X=POKE
038F A9        114          HEX A9              ;Y=RETURN
0390 3B        115          HEX 3B              ;Z=PEEK
0391           116          END

***** END OF ASSEMBLY
```

TYPE

Type is a game designed to increase your typing skills on the Apple. It clears the screen and flashes a letter on the screen in a random place. You are given a certain amount of time in which to press that key (time is selected at the beginning of the program by the user). If you do not press the key within that time you go onto the next key (10 to 50 letters, selected by you at the beginning of the program). If you press the incorrect key, you are not penalised but must still press the correct key.

This program could easily be adapted for use on other micros. The statement in line 1600 simply clicks the speaker. The timing may have to be adjusted on faster or slower micros (this was done on a IIe). This is in the for-next loops.

The statement in 1300 simply gets a character or checks if one has been pressed. It can be changed to an 'INKEY$' statement. 'Inverse' makes all characters printed after it appear black on white (instead of white on black) until the 'Normal' statement.

All the rems can be omitted.

Tony Humfrey
Parkes NSW

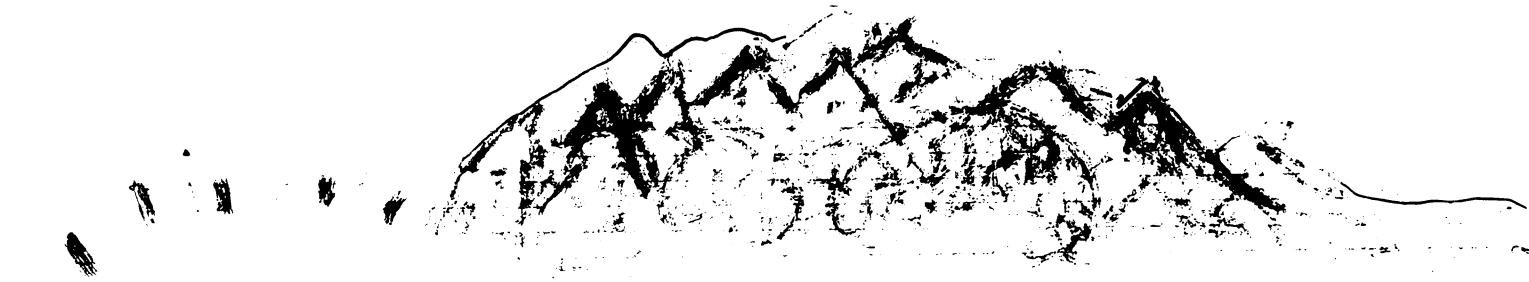
```
 1  REM _______________
 2  REM !             !
 3  REM !     TYPE    !
 4  REM !             !
 5  REM ! BY T.Humfrey !
 6  REM !             !
 7  REM !_____________!
 8  REM
 9  REM ____________________
10  REM  Member of the
11  REM
12  REM  Parkes High School
13  REM
14  REM  COMPUTER-
15  REM
16  REM            CLUB
17  REM
18  REM ____________________
100  HOME : ONERR  GOTO 2700
200  INPUT "No. OF GAMES(10 TO 50
     )->";GAMES
300  IF GAMES < 10 OR GAMES > 50 THEN
     100
400  PRINT : PRINT : PRINT : PRINT
     : PRINT : INPUT "SPEED (0.5
     SECONDS TO 5 SECONDS)   ->";
     SPD: IF SPD < 0.5 OR SPD > 5
      THEN 400
500  FOR G = 1 TO GAMES
600  HOME
700 V =  INT ( RND (1) * 24) + 1
800 H =  INT ( RND (1) * 40) + 1
900 C =  INT ( RND (1) * 61) + 34
1000 C$ =  CHR$ (C)
1100  VTAB V: HTAB H: PRINT C$
1200  FOR A = 1 TO SPD * 100
1300  IF  PEEK ( - 16384) > 127 THEN
      GET R$
1400  IF R$ = C$ THEN 2000
1500  NEXT A
1600  FOR B = 1 TO 100
1700 A =  PEEK ( - 16336)
1800  NEXT B
1900  NEXT G
1950  GOTO 2250
2000  PRINT "HIT": FOR T = 1 TO 4
     00: NEXT T
2100 I = I + 1
2200  NEXT G
2250  HOME : INVERSE : VTAB 1: HTAB
     14: PRINT "SPEED ";SPD;"C/P/
     S": NORMAL
2300  VTAB 11: PRINT "YOU GOT ";I
     ;" OUT OF ";GAMES;" RIGHT": PRINT

2400  IF I =  > GAMES - (GAMES /
     5) THEN  PRINT "GREAT GOING!
     ": FOR F = 1 TO 1000: NEXT :
      GOTO 2650
2500  IF I < GAMES - (GAMES / 5) AND
     I =  > GAMES / 2 THEN  PRINT
     "FAIRLY GOOD": FOR F = 1 TO
     1000: NEXT : GOTO 2650
2600  IF I < GAMES / 2 THEN  PRINT
     "NEED PRACTICE": FOR F = 1 TO
     1000: NEXT : GOTO 2650
2700  PRINT "ANOTHER GAME(Y/N)";:
      GET Y$: IF Y$ = "Y" THEN  GOTO
     100: IF Y$ = "N" THEN  END :
      GOTO 2700

12800 REM INSERT CTRL◆-G IN LINE 2000
```

LORD OF THE RINGS

This is a program written for a 48K Apple II+ aimed at restoring peace to Middle Earth. It is a long listing and to save typing it out, I will put it on disk for everyone who sends me a disk and five dollars. All proceeds will go to the Parkes High School Computer Club. Please mark all parcels "Computer disks – keep away from magnets". Write care of Your Computer and it will be passed on.

Shaun Humfrey
Parkes NSW

```
0  TEXT : HOME
2  REM   LORD OF THE RINGS BY SHAUN HUMFREY
5  HOME : CLEAR
7 M = 0
10  HTAB 12: INVERSE : PRINT "LORD OF THE RINGS": NORMAL
20  VTAB 5: PRINT : VTAB 5: INPUT "ENTER NAME ->";N$
35  IF N$ = "LEGOLAS" THEN 75
37  IF N$ = "BILBO" THEN 80
40  IF N$ = "BOROMIR" THEN 85
42  IF N$ = "SAURON" THEN 5
44  IF N$ = "GANDALF" THEN 5
45  IF N$ = "THEODEN" THEN 5
46  IF N$ = "ARAGORN" THEN 5
47  IF N$ = "FRODO" THEN 5
48  IF N$ = "GALADRIEL" THEN 5
49  GOTO 100
50 R$ = "ELF":RS$ = "ELVES":W$ = "BOW": GOTO 200
55 R$ = "DWARF":RS$ = "DWARVES":W$ = "AXE": GOTO 200
57 R$ = "HOBBIT":RS$ = "HOBBITS":W$ = "KNIFE": GOTO 200
60 R$ = "HUMAN":RS$ = "HUMANS":W$ = "SWORD": GOTO 200
70  PRINT : PRINT "GREETINGS GIMLI,SON OF GLOIN.": FOR I = 1 TO 2000: NEXT I: GO
TO
     55
75  PRINT : PRINT "I WISH THEE WELL,LEGOLAS.": FOR I = 1 TO 2000: NEXT I: GOTO 5
0
80  PRINT : PRINT "GREETINGS BILBO,FINDER OF THE RING.": FOR I = 1 TO 2000: NEXT
 I
     : GOTO 57
85  PRINT : PRINT "GOOD LUCK ON YOUR QUEST,BOROMIR.": FOR I = 1 TO 2000: NEXT I:
 GOTO
     60
100  PRINT : PRINT : PRINT "(1) DWARF": PRINT : PRINT "(2) ELF": PRINT : PRINT "
(3
     ) HUMAN": PRINT : PRINT "(4) HOBBIT": PRINT : PRINT : PRINT
110  PRINT "ENTER RACE ->";: GET R
120  IF R > 4 THEN 100
126  IF R < 1 THEN 100
130  ON R GOTO 55,50,60,57
199  REM  INSTRUCTIONS
200  HOME
210  PRINT "  WELCOME TO RIVENDELL,";N$;".": PRINT : PRINT "YOU HAVE BEEN CHOSEN
 T
     O REPRESENT ": PRINT : PRINT RS$;" IN THE COMPANY SELECTED TO ": PRINT : PR
INT
     "DESTROY THE RING OF SAURON.": PRINT : PRINT : PRINT "DO YOU WANT MORE INFO
RM
     ATION ABOUT THE": PRINT : PRINT "THE RING ? ";: GET A$
220  IF A$ = "Y" THEN 250
225  IF A$ = "N" THEN 400
230  GOTO 200
249  REM  ABOUT THE RING
250  HOME : PRINT " THE RINGS OF POWER WERE FORGED IN THE": PRINT : PRINT "CRACK
 O
     F DOOM BY SAURON,EVIL LORD OF": PRINT : PRINT "MORDOR.THESE RINGS CORRUPT T
HE
      SPIRIT": PRINT : PRINT "AND DECAY THE BODY.KNOWING THIS HE": PRINT : PRINT
 "
     GAVE NINE RINGS TO MEN,SEVEN TO DWARVES": PRINT
260  PRINT "AND KEPT THE RULING RING TO CONTROL THE": PRINT : PRINT "OTHERS.THRE
E
     GOOD RINGS WERE MADE BY": PRINT : PRINT "ELVES,BUT THEY TOO ARE AFFECTED BY
 T
```



LORD OF THE RINGS

```
     HE": PRINT : PRINT "ONE RULING RING."
270  PRINT : PRINT " MEN AND DWARVES WITH THE RINGS FELL": PRINT : PRINT "INTO T
HE
      SERVICE OF SAURON."
275  VTAB 24: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
277  HOME : PRINT "THE NINE MEN WITH RINGS WORSHIPED": PRINT : PRINT "SAURON.ONL
Y
     THREE OF THE SEVEN ": PRINT : PRINT "DWARF-LORDS CAME TO SAURON.THE OTHERS"
: PRINT
     : PRINT "WERE KILLED BY DRAGONS OR DEMONS."
279  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
280  HOME : PRINT " SAURON LOST THE RULING RING IN BATTLE": PRINT : PRINT "AGAIN
ST
      A RACE OF MEN FROM ACROSS THE": PRINT : PRINT "GREAT SEA.CENTURIES LATER I
T
     WAS": PRINT : PRINT "FOUND BY A SMALL BEING CALLED SMEAGOL.": PRINT : PRINT
 "
     HE EVENTUALLY LOST IT AND IT WAS ": PRINT
290  PRINT "LATER FOUND BY A HOBBIT NAMED BILBO.HIS": PRINT : PRINT "NEPHEW,FROD
O,
     NOW HAS IT.FRODO IS IN": PRINT : PRINT "YOUR COMPANY AS RINGBEARER."
300  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
400  HOME
410  PRINT "ALSO IN YOUR PARTY ARE GANDALF,A ": PRINT : PRINT "POWERFUL WIZARD;A
RA
     GORN,RIGHTFUL KING ": PRINT : PRINT "OF GONDOR;AND FRODO,THE RINGBEARER."
420  VTAB 21: PRINT "PRESS ANY KEY TO EMBARK ON YOUR": PRINT : PRINT "JOURNEY";:
 GET
     A$
499  REM  THE BEGINNING!!!
500  HOME
510  PRINT " YOU EMBARK ON A COLD WINTER MORNING.": PRINT : PRINT "YOU WALK MANY
 L
     EAGUES UNTIL"
520  PRINT : PRINT "YOU REACH THE ANCIENT AND LONG ": PRINT : PRINT "ABANDONED D
WA
     RF-KINGDOM OF MORIA WHICH": PRINT : PRINT "FILLS THE INSIDE OF A MOUNTAIN."
530  PRINT : PRINT "HERE YOU HAVE A CHOICE...WILL YOU GO": PRINT : PRINT "THROUG
H
     THE REALMS OF MORIA OR GO OVER": PRINT : PRINT "THE MOUNTAIN VIA A SMALL PA
TH
      ? "
540  PRINT : PRINT : PRINT "(ENTER M FOR MORIA AND P FOR PATH.) ";: GET D$
545  PRINT
550  IF D$ = "P" THEN 600
555  IF D$ = "M" THEN 800
560  GOTO 540
599  REM  MOUNTAIN PATH
600  HOME : PRINT " AS YOU ASCEND THE MOUNTAIN A SNOWSTORM": PRINT : PRINT "STAR
TS
     .THIS COULD BE WORSE THAN YOU": PRINT : PRINT "THOUGHT."
610  PRINT : PRINT : PRINT "ARE YOU SURE YOU WANT TO GO ON ? ";: GET A$
615  PRINT
620  IF A$ = "N" THEN 520
625  IF A$ = "Y" THEN 650
630  GOTO 610
650 B =  INT ( RND (1) * 10) + 1: IF B < 3 THEN 670
660  PRINT : PRINT "THE SNOWSTORM DIES DOWN,AND YOU ": PRINT : PRINT "CONTINUE U
P
     THE PATH": PRINT : PRINT : PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE
";
     : GET A$
662 M = M + 300
665  GOTO 680
670  PRINT : PRINT "THE BLIZZARD CONTINUES AND YOU FREEZE": PRINT : PRINT "TO DE
AT
```

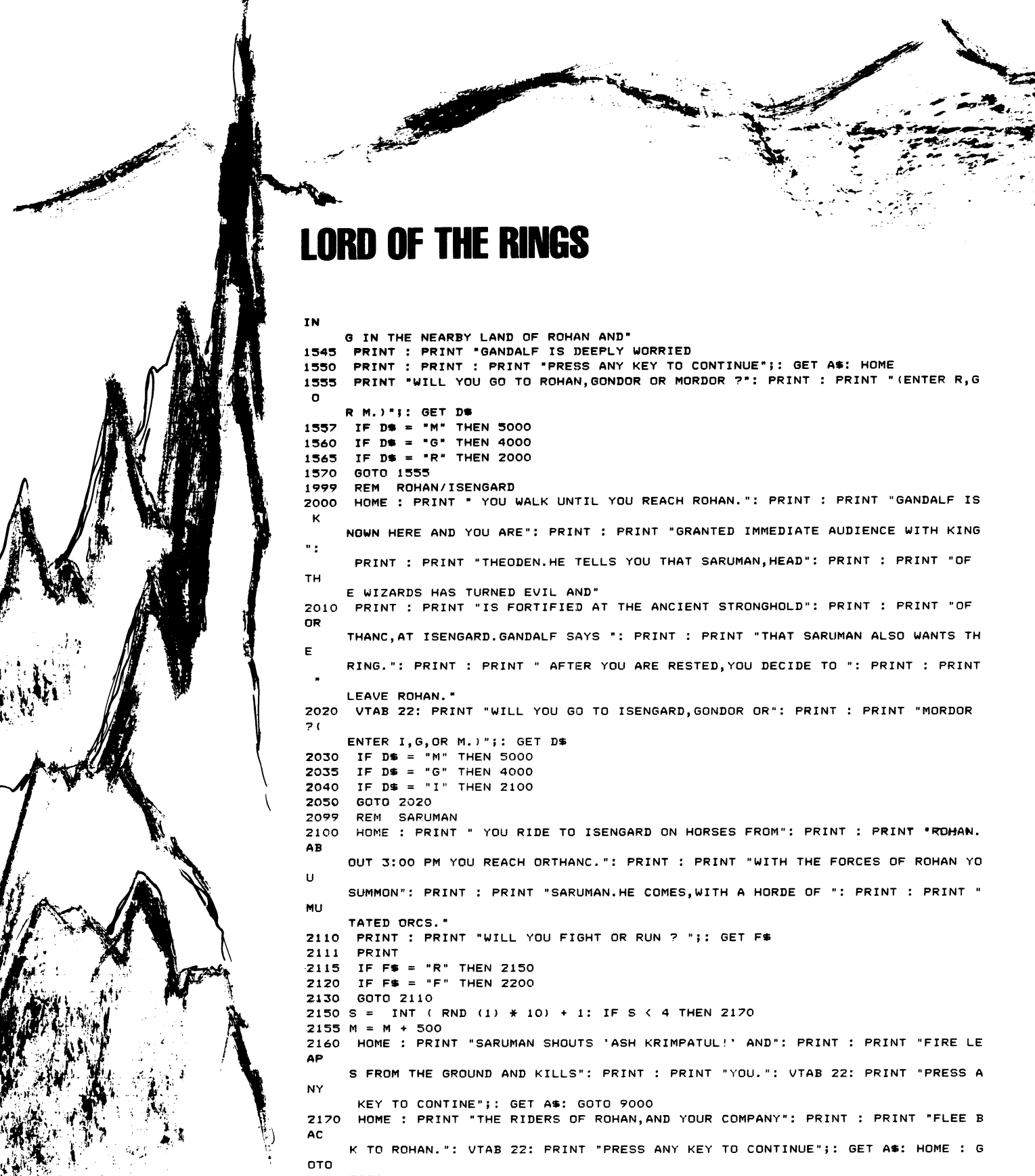
```
     H.": PRINT : PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GO
TO
     9000
680  HOME : PRINT "YOU PROCEED UNTIL YOU COME TO A FORK": PRINT : PRINT "IN THE
RO
     AD."
690  PRINT : PRINT "WILL YOU GO LEFT OR RIGHT ? ";: GET D$
700  IF D$ = "L" THEN 720
705  IF D$ = "R" THEN 750
710  GOTO 690
720  HOME : PRINT "YOU TAKE THE LEFT PATH.SOON YOU HEAR A": PRINT : PRINT "DISTA
NT
      RUMBLE.YOU LOOK UP TO SEE": PRINT : PRINT "TONNES OF ROCK FALLING TOWARDS
YO
     U.": PRINT : PRINT "YOUR PARTY HAS BEEN KILLED IN AN": PRINT : PRINT "AVALA
NC
     E."
730  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 9000
750  HOME : PRINT "YOU TAKE THE RIGHT PATH.YOU WALK DOWN": PRINT : PRINT "THE OT
HE
     R SIDE OF THE MOUNTAIN SAFELY."
770  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
790  GOTO 1100
799  REM  MORIA
800  HOME
810  PRINT "YOU ENTER MORIA SLOWLY.IT IS DARK AND": PRINT : PRINT "THERE IS A SE
NS
     E OF EVIL IN THE AIR.": PRINT : PRINT "THIS COULD BE WORSE THAN YOU THOUGHT
."

820  PRINT : PRINT "ARE YOU SURE YOU WANT TO GO ON ? ";: GET A$
822  PRINT
825  IF A$ = "N" THEN 520
830  IF A$ = "Y" THEN 850
840  GOTO 820
850  PRINT : PRINT "GANDALF EMITS A GLOW FROM THE END OF ": PRINT : PRINT "HIS S
TA
     FF.YOU CAN SEE SIDE PASSAGES TO": PRINT : PRINT "THE LEFT AND RIGHT."
860  PRINT : PRINT "DO YOU WANT TO EXPLORE A SIDE PASSAGE ?": GET A$
862  PRINT
865  IF A$ = "Y" THEN 890
870  IF A$ = "N" THEN 1000
875  IF A$ = "L" THEN 900
880  IF A$ = "R" THEN 950
885  GOTO 860
890  PRINT : PRINT "LEFT OR RIGHT ? ";: GET A$
891  PRINT
892  IF A$ = "L" THEN 900
894  IF A$ = "R" THEN 950
895  GOTO 890
899  REM  ORC DOOR
900  HOME : PRINT "YOU WALK DOWN THE PASSAGE AND COME TO A": PRINT : PRINT "A DO
OR
     ."
905  PRINT : PRINT "WILL YOU OPEN IT ? ";: GET A$: PRINT
910  IF A$ = "N" THEN 920
912  IF A$ = "Y" THEN 925
915  GOTO 905
920  PRINT : PRINT "YOU LEAVE THE DOOR AND COME BACK TO THE": PRINT : PRINT "MAI
N
     HALLWAY.": FOR I = 1 TO 3500: NEXT I: GOTO 1000
925  HOME : PRINT "YOU BREAK THROUGH THE DOOR AND ARE": PRINT : PRINT "IMMEDIATE
LY
      CONFRONTED BY A BAND OF ": PRINT : PRINT "ORCS."
927  PRINT : PRINT "WILL YOU FIGHT OR RUN ? ";: GET F$
928  PRINT
```

▷

LORD OF THE RINGS

```
929  IF F$ = "R" THEN 945: IF F$ = "F" THEN 930: GOTO 927
930  HOME : PRINT "ARAGORN DRAWS HIS SWORD AND HEWS AT THE": PRINT : PRINT "ORCS
.L
     IGHTNING LEAPS FROM GANDALF'S": PRINT : PRINT "STAFF,KILLING MANY ORCS.YOU
WI
     ELD YOUR": PRINT : PRINT W$;" SKILLFULLY."
932 K =  INT ( RND (1) * 25) + 1:S =  INT ( RND (1) * 10) + 1
934  PRINT : PRINT "DURING BATTLE YOU FIGHT VALIANTLY AND": PRINT : PRINT "KILL
";
     K;" ORCS."
935  IF S < 3 THEN 946
937  PRINT : PRINT "EVENTUALLY,YOU KILL ALL THE ORCS.": VTAB 22: PRINT "PRESS AN
Y
     KEY TO CONTINUE";: GET A$
938  PRINT : PRINT
939 M = M + 800: GOTO 860
945  HOME :S =  INT ( RND (1) * 10) + 1: IF S < 4 THEN 947
946  PRINT "YOU ARE ALL SLAUGHTERED BY THE ORCS": FOR I = 1 TO 3000: NEXT I: GOT
O
     9000
947  PRINT "YOU RUN DOWN THE TUNNEL BACK TO THE": PRINT : PRINT "MAIN HALLWAY":
FOR
     I = 1 TO 3000: NEXT I: GOTO 1000
950  HOME : PRINT "YOU ARE CONFRONTED BY A FIRE DEMON,A": PRINT : PRINT "BALROG.
":
      PRINT : PRINT "WILL YOU FIGHT OR RUN ? ";: GET F$
951  PRINT
952  IF F$ = "F" THEN 960
954  IF F$ = "R" THEN 957
955  GOTO 950
957 C =  INT ( RND (1) * 10) + 1
958  IF C < 3 THEN 947
959  PRINT : PRINT "THE BALROG CASTS A SPELL,AND YOU CANT": PRINT : PRINT "LEAVE
 T
     HE ROOM."
960  PRINT : PRINT "YOU ATTACK THE BALROG WITH YOUR ";W$: PRINT : PRINT "ARAGORN
 L
     EAPS AT THE BALROG'S THROAT."
962  IF F$ = "R" THEN 965
963  GOTO 970
965  PRINT : PRINT 'THE BALROG CASTS GANDALF INTO AN ABYSS.":GA$ = "DEAD"
967  GOTO 972
970 GA =  INT ( RND (1) * 10) + 1
971  IF GA < 3 THEN 965
972 S =  INT ( RND (1) * 10) + 1
975  IF S < 4 THEN 980
977  GOTO 984
980  PRINT : PRINT "THE BALROG FIGHTS LIKE A DEMON (WHICH": PRINT : PRINT "IT IS
)
     AND KILLS YOU ALL.": PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GE
T
     A$: GOTO 9000
984  PRINT : PRINT : PRINT
985  PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: HOME : PRINT "WITH YOUR
 "
     ;W$;" YOU WOUND THE": PRINT : PRINT "BALROG IN THE THROAT,KILLING IT."
986 M = M + 600
987  PRINT : PRINT "ON THE FLOOR YOU FIND A RING.": IF GA$ = "DEAD" THEN 990
989  PRINT : PRINT "GANDALF SAYS IT IS ONE OF THE LOST": PRINT : PRINT "RINGS OF
 P
     OWER OF THE DWARF-LORDS."
990  PRINT : PRINT "THE RING IS ONLY TO BE USED IN EXTREME": PRINT : PRINT "EMER
GE
     NCIES.": VTAB 22
992 I$ = "RING"
995  PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
```

```
1000  HOME : PRINT "YOU NEAR THE EXIT TO MORIA.": PRINT : PRINT "SUDDENLY YOU HE
AR
      THE BOOM OF DISTANT ": PRINT : PRINT "DRUMS AND ORC ISSUE FORTH FROM THE":
 PRINT
     : PRINT "EASTERN DOOR."
1010  IF GA$ = "DEAD" THEN 1050
1020  PRINT : PRINT "GANDALF CASTS A SPELL AND THE EASTERN": PRINT : PRINT "DOOR
 A
     ND NEARBY CEILING COLLAPSE": PRINT : PRINT "KILLING THE ORCS.": VTAB 22: PR
INT
     "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 1100
1050  PRINT : PRINT "WILL YOU FIGHT OR RUN ? ";: GET F$
1052  PRINT
1055  IF F$ = "R" THEN 1080: IF F$ = "F" THEN 1060: GOTO 1050
1060  HOME : PRINT "YOU AND ARAGORN FIGHT SIDE BY SIDE": PRINT : PRINT "KILLING
MA
     NY ORCS.YOUR ";W$;" IS A": PRINT : PRINT "GOOD WEAPON."
1065 S =  INT ( RND (1) * 10) + 1:K =  INT ( RND (1) * 20) + 1
1070  PRINT : PRINT "YOU KILL ";K;" ORCS WITH YOUR ";W$
1075  IF S < 4 THEN 1079
1076 M = M + 600
1077  PRINT : PRINT "YOU FINALLY KILL ALL THE ORCS.": PRINT : PRINT : PRINT : PR
INT
     "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 1100
1079  PRINT : PRINT "YOU FIGHT VALIANTLY BUT SOON TIRE.": GOTO 1090
1080 S =  INT ( RND (1) * 10) + 1: IF S < 3 THEN 1100
1090  PRINT : PRINT "THE ORCS KILL YOU ALL.": PRINT : PRINT : PRINT : PRINT "PRE
SS
      ANY KEY TO CONTINUE";: GET A$: GOTO 9000
1100  HOME : PRINT "YOU REST AT THE FOOT OF THE MOUNTAIN TO": PRINT : PRINT "PLA
N
     YOUR NEXT MOVE."
1110  PRINT : PRINT "WILL YOU GO THROUGH THE STRANGE FOREST": PRINT : PRINT "OF
LO
     THLORIEN TO GET TO GONDOR TO GET": PRINT : PRINT "HELP,OR GO STRAIGHT TO MO
RD
     OR ?"
1120  PRINT : PRINT "(ENTER G FOR GONDOR,M FOR MORDOR.)";: GET D$
1122  PRINT
1125  IF D$ = "M" THEN 5000
1130  IF D$ = "G" THEN 1500
1140  GOTO 1120
1499  REM  LOTHLORIEN
1500  HOME
1505  PRINT "STRANGE TALES ARE TOLD ABOUT THE FOREST": PRINT : PRINT "OF LOTHLOR
IE
     N."
1510  PRINT : PRINT "ARE YOU SURE YOU WANT TO GO ON ? ";: GET A$
1511  PRINT
1512  IF A$ = "N" THEN 1100
1515  IF A$ = "Y" THEN 1520
1517  GOTO 1500
1520  HOME : PRINT "ELVISH WARRIORS CAPTURE YOU AND TAKE ": PRINT : PRINT "YOU T
O
     THEIR QUEEN,GALADRIEL."
1525  IF GA$ = "DEAD" THEN 1530
1528  GOTO 1540
1530  PRINT : PRINT "GANDALF IS ALSO THERE.APPARENTLY,HE ": PRINT : PRINT "SURVI
VE
     D THE BALROG.HE LOOKS WISE AND ": PRINT : PRINT "MORE DISTANT THAN BEFORE."
1535 GA$ = ""
1540  PRINT : PRINT "GALADRIEL OFFERS YOU FOOD AND REST": PRINT : PRINT "WHICH Y
OU
      GRATEFULLY ACCEPT.SEVERAL": PRINT : PRINT "DAYS LATER YOU DECIDE TO LEAVE.
":
      PRINT : PRINT "GALADRIEL WARNS THAT THERE IS TROUBLE": PRINT : PRINT "BREW
```



LORD OF THE RINGS

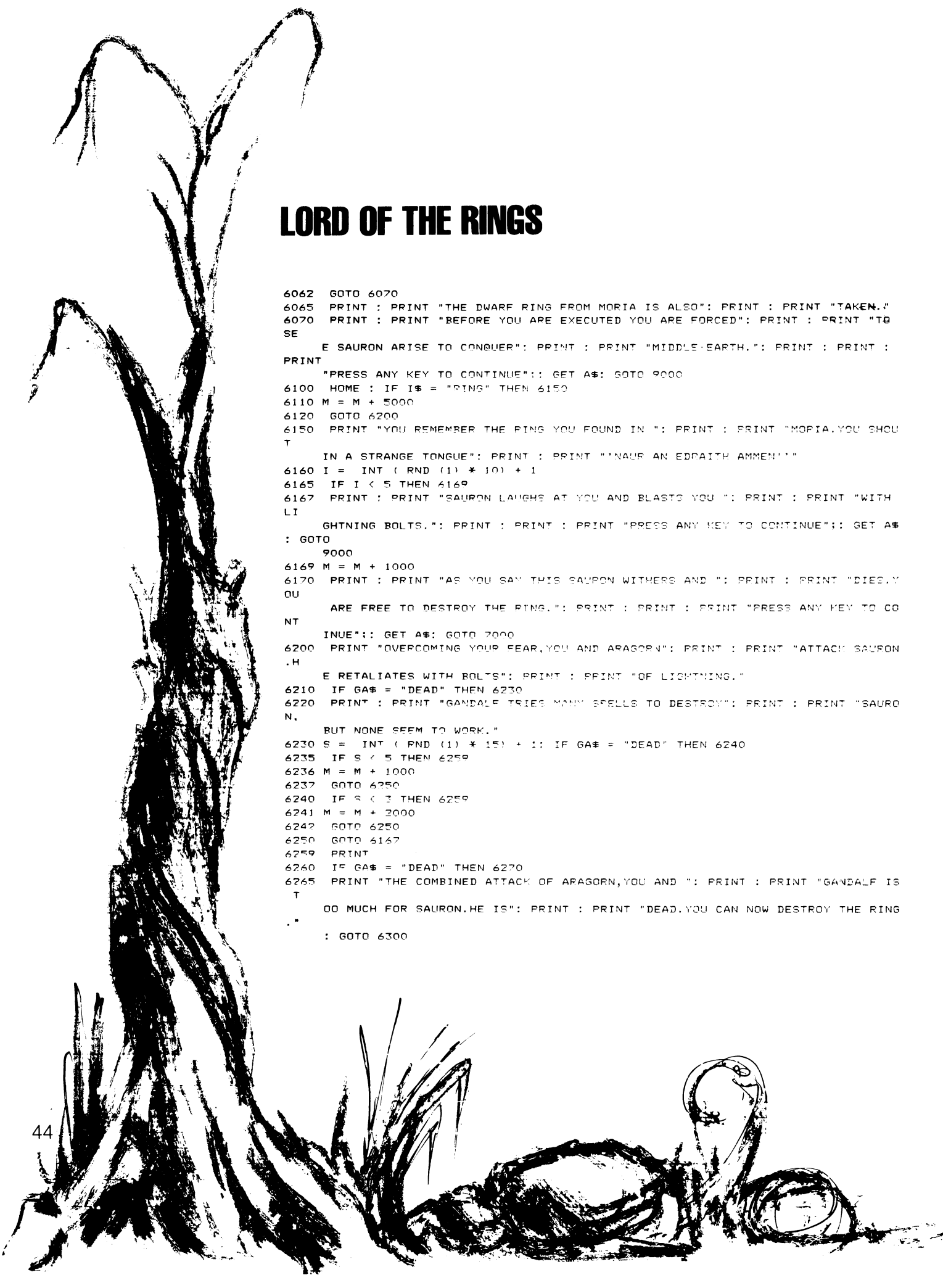
```
IN
     G IN THE NEARBY LAND OF ROHAN AND"
1545  PRINT : PRINT "GANDALF IS DEEPLY WORRIED
1550  PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: HOME
1555  PRINT "WILL YOU GO TO ROHAN,GONDOR OR MORDOR ?": PRINT : PRINT "(ENTER R,G
 O
     R M.)";: GET D$
1557  IF D$ = "M" THEN 5000
1560  IF D$ = "G" THEN 4000
1565  IF D$ = "R" THEN 2000
1570  GOTO 1555
1999  REM  ROHAN/ISENGARD
2000  HOME : PRINT " YOU WALK UNTIL YOU REACH ROHAN.": PRINT : PRINT "GANDALF IS
 K
     NOWN HERE AND YOU ARE": PRINT : PRINT "GRANTED IMMEDIATE AUDIENCE WITH KING
":
      PRINT : PRINT "THEODEN.HE TELLS YOU THAT SARUMAN,HEAD": PRINT : PRINT "OF
TH
     E WIZARDS HAS TURNED EVIL AND"
2010  PRINT : PRINT "IS FORTIFIED AT THE ANCIENT STRONGHOLD": PRINT : PRINT "OF
OR
     THANC,AT ISENGARD.GANDALF SAYS ": PRINT : PRINT "THAT SARUMAN ALSO WANTS TH
E
     RING.": PRINT : PRINT " AFTER YOU ARE RESTED,YOU DECIDE TO ": PRINT : PRINT
 "
     LEAVE ROHAN."
2020  VTAB 22: PRINT "WILL YOU GO TO ISENGARD,GONDOR OR": PRINT : PRINT "MORDOR
?(
     ENTER I,G,OR M.)";: GET D$
2030  IF D$ = "M" THEN 5000
2035  IF D$ = "G" THEN 4000
2040  IF D$ = "I" THEN 2100
2050  GOTO 2020
2099  REM  SARUMAN
2100  HOME : PRINT " YOU RIDE TO ISENGARD ON HORSES FROM": PRINT : PRINT "ROHAN.
AB
     OUT 3:00 PM YOU REACH ORTHANC.": PRINT : PRINT "WITH THE FORCES OF ROHAN YO
U
     SUMMON": PRINT : PRINT "SARUMAN.HE COMES,WITH A HORDE OF ": PRINT : PRINT "
MU
     TATED ORCS."
2110  PRINT : PRINT "WILL YOU FIGHT OR RUN ? ";: GET F$
2111  PRINT
2115  IF F$ = "R" THEN 2150
2120  IF F$ = "F" THEN 2200
2130  GOTO 2110
2150 S =  INT ( RND (1) * 10) + 1: IF S < 4 THEN 2170
2155 M = M + 500
2160  HOME : PRINT "SARUMAN SHOUTS 'ASH KRIMPATUL!' AND": PRINT : PRINT "FIRE LE
AP
     S FROM THE GROUND AND KILLS": PRINT : PRINT "YOU.": VTAB 22: PRINT "PRESS A
NY
      KEY TO CONTINE";: GET A$: GOTO 9000
2170  HOME : PRINT "THE RIDERS OF ROHAN,AND YOUR COMPANY": PRINT : PRINT "FLEE B
AC
     K TO ROHAN.": VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: HOME : G
OTO
     2020
2200  HOME : PRINT "YOU LEAD THE FORCES OF ROHAN INTO": PRINT : PRINT "BATTLE.YO
U
     KILL "; INT ( RND (1) * 15) + 1;" ORCS WITH YOUR": PRINT : PRINT W$;"."
2210 S =  INT ( RND (1) * 10) + 1: IF S < 6 THEN 2250
2220  PRINT : PRINT "SARUMANS ORCS KILL ALL OF YOU.": VTAB 22: PRINT "PRESS ANY
KE
     Y TO CONTINUE";: GET A$: GOTO 9000
2250  PRINT : PRINT "GANDALF FIGHTS SARUMAN AND DESTROYS HIM.": PRINT : PRINT "T
```

```
HE
     SURVIVING ORCS IMMEDIATELY ": PRINT : PRINT "SURRENDER."
2252 M = M + 1000
2255  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: HOME
2260  PRINT " YOU ENTER SARUMAN'S ROOM IN SEARCH OF": PRINT : PRINT "ANYTHING US
EF
    UL.ON THE DESK YOU SEE ": PRINT : PRINT "A GLASS SPHERE."
2265  PRINT : PRINT "WILL YOU GET IT ? ";: GET A$
2267  PRINT
2270  IF A$ = "Y" THEN 2300
2280  IF A$ = "N" THEN 2400
2290  GOTO 2265
2300  HOME : PRINT "YOU GET THE SPHERE AND SHOW IT TO ": PRINT : PRINT "GANDALF.
HE
     SAYS IT IS A SEEING STONE,A": PRINT : PRINT "PALANTIR,OR BASICALLY A CRYST
AL
     BALL."
2305 P$ = "SPHERE"
2310  PRINT : PRINT "WILL YOU USE IT ? ";: GET A$
2315  PRINT
2320  IF A$ = "Y" THEN 2350
2325  IF A$ = "N" THEN 2400
2330  GOTO 2310
2350 P =  INT ( RND (1) * 10) + 1: IF P < 4 THEN 2370
2352 M = M + 200
2355  HOME : PRINT "YOU GAZE INTO THE PALANTIR AND SEE": PRINT : PRINT "A GIANT
SP
    IDER IN A SMALL TUNNEL.THIS": PRINT : PRINT "VISION FADES AND IS REPLACED B
Y
    THE": PRINT : PRINT "CRACK OF DOOM.BEFORE IT IS SAURON.HE IS": PRINT : PRIN
T
    "WAITING FOR YOU."
2360  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 2400
2370  HOME : PRINT "YOU LOOK INTO THE PALANTIR AND SEE": PRINT : PRINT "SAURON.H
E
    SEES YOU AND SUDDENLY A LIGHT": PRINT : PRINT "STABS OUT OF THE PALANTIR AN
D
    KILLS YOU": VTAB 22
2375  PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 9000
2400  HOME : PRINT "YOU RIDE BACK TO ROHAN,THINKING OF THE": PRINT : PRINT "DAY'
S
    EVENTS.YOU FINALLY DECIDE TO ": PRINT : PRINT "LEAVE ROHAN."
2410  VTAB 22: PRINT "WILL YOU GO TO GONDOR OR MORDOR ? ";: GET D$
2420  IF D$ = "M" THEN 5000
2425  IF D$ = "G" THEN 4000
2440  GOTO 2410
3999  REM GONDOR
4000  HOME
4010  HOME : PRINT "YOU ARRIVE AT GONDOR IN THE EVENING.YOU": PRINT : PRINT "ARE
 G
    RANTED AN AUDIENCE WITH THE": PRINT : PRINT "STEWARD OF GONDOR.HERE ARAGORN
 P
    UTS ": PRINT : PRINT "FORTH HIS CLAIM TO THE THRONE OF ": PRINT : PRINT "GO
ND
    OR."
4020 B =  INT ( RND (1) * 10) + 1: IF B < 5 THEN 4050
4025 M = M + 500
4030  PRINT : PRINT : PRINT "HE IS DISBELIEVED AND YOU ARE BANISHED": PRINT : PR
INT
     "FROM THE KINGDOM.YOU DECIDE TO GO TO": PRINT : PRINT "MORDOR WITHOUT THE A
ID
     OF GONDOR."
4040  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 5000
4050  PRINT : PRINT "THE STEWARD KNEELS AND PLEDGES HIS": PRINT : PRINT "ALLEGIE
NC
    E TO ARAGORN.": PRINT : PRINT " DAYS LATER.GONDOR IS BESEIGED.THE": PRINT :
```

LORD OF THE RINGS

```
 PRINT
     "FORCES OF SAURON HAVE GATHERED.THERE": PRINT : PRINT "ARE MILLIONS OF ORCS
,S
     OME WARGS (GIANT": PRINT : PRINT "WOLVES),AND SOME DRAGONS.": PRINT : PRINT

4060  PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
4065  HOME : PRINT "WILL YOU FIGHT OR SURRENDER ? ";: GET F$
4070  IF F$ = "F" THEN 4100
4080  IF F$ = "S" THEN 6050
4090  GOTO 4065
4100  HOME : PRINT "YOU HELP DEFEND GONDOR.A DRAGON SWOOPS": PRINT : PRINT "LOW
AN
     D YOU KILL IT WITH YOUR ";W$;".": PRINT : PRINT "YOU KILL MANY WARGS AND OR
CS
      BUT YOU": PRINT : PRINT "ARE TIRING."
4105 M = M + 700
4110 S =  INT ( RND (1) * 10) + 1: IF S < 7 THEN 4150
4120  PRINT : PRINT "GONDOR IS SOON LAID IN RUBBLE BY THE": PRINT : PRINT "INVAD
ER
     S.A WARG LEAPS AT YOUR THROAT.": PRINT : PRINT "YOU ARE DEAD."
4130  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
4140  GOTO 9000
4150  PRINT : PRINT "YOU CAPTURE THE LEADER OF THE INVASION.": PRINT : PRINT "IT
 I
     S A HUMAN WARRIOR.YOU TELL HIM TO": PRINT : PRINT "MAKE HIS FORCES RETREAT
AN
     D THEY DO.": PRINT : PRINT "YOU'VE WON THE BATTLE!!"
4160  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: HOME
4165  IF P$ = "SPHERE" THEN 4170
4166  GOTO 4200
4170  PRINT "BACK IN THE THRONE ROOM YOU DECIDE TO": PRINT : PRINT "USE THE PALA
NT
     IR."
4172 M = M + 200
4175 P =  INT ( RND (1) * 10) + 1: IF P < 3 THEN 4180
4177  PRINT : PRINT "YOU SEE WHOLE ARMIES STILL LEFT IN ": PRINT : PRINT "MORDOR
."

4178  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE": GET A$
4179  GOTO 4200
4180  PRINT : PRINT "YOU SEE SAURON.HE SEES YOU AND MUTTERS": PRINT : PRINT "SOM
ET
     HING.IMMEDIATELY YOU ARE DEAD."
4190  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 9000
4200  HOME : PRINT "YOU DECIDE THAT A SMALL PARTY IS ALL": PRINT : PRINT "THAT C
AN
      BE USED TO DESTROY THE RING.": PRINT : PRINT "THE ORIGINAL PARTY SETS OUT
TO
      MORDOR.": PRINT : PRINT "THE STEWARD RULES GONDOR AGAIN."
4210  VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
4999  REM  MORDOR
5000  HOME : PRINT " YOU HAVE ARRIVED AT THE GATES OF : PRINT : PRINT "MORDOR.WI
LL
      YOU GET IN THROUGH THE ": PRINT : PRINT "GATES OR THROUGH A TUNNEL AT THE"
: PRINT
     : PRINT "WESTERN EDGE OF MORDOR ?"
5010  PRINT : PRINT "(ENTER G FOR GATE,T FOR TUNNEL.)";: GET D$
5011  PRINT
5015  IF D$ = "G" THEN 5050
5020  IF D$ = "T" THEN 5200
5030  GOTO 5010
5050  IF GA$ = "DEAD" THEN 5060
5055  GOTO 5070
5060  PRINT : PRINT "HOW CAN WE GET THROUGH LOCKED GATES": PRINT : PRINT "WITHOU
T
     GANDALF ?!": FOR I = 1 TO 5000: NEXT I: GOTO 5010.
```

```
5070  PRINT : PRINT "GANDALF CASTS A SPELL AND THE GATES": PRINT : PRINT "ARE DE
MO
     LISHED."
5075 M = M + 2000
5080 N =  INT ( RND (1) * 10) + 1: IF N < 2 THEN 5100
5090  PRINT : PRINT "THROUGH THE GATES RIDES THE NAZGUL.THE": PRINT : PRINT "NIN
E
     MEN WITH RINGS OF POWER.THEIR": PRINT : PRINT "CAPTAIN SCREAMS AND CHARGES
AT
      THE ": PRINT : PRINT "COMPANY.THE OTHER EIGHT FOLLOW AND THEY": PRINT : PR
INT
     "SLAY THE WHOLE PARTY.": PRINT : PRINT : PRINT : PRINT : PRINT "PRESS ANY K
EY
      TO CONTINUE";
5095  GET A$: GOTO 9000
5100  PRINT : PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 6
00
     0
5200  HOME : PRINT "YOU TRAVEL UP A DISUSED TUNNEL LEADING": PRINT : PRINT "TO T
HE
      CRACK OF DOOM.SUDDENLY AN ": PRINT : PRINT "ENORMOUS SPIDER LEAPS OUT OF T
HE
      DARK": PRINT : PRINT "AT YOU.LEGEND SAYS THE SPIDER IS CALLED": PRINT : PR
INT
     "'SHELOB' AND IS EXTREMELY DANGEROUS."
5210  PRINT : PRINT "WILL YOU FIGHT OR RUN ? ";: GET F$
5211  PRINT
5215  IF F$ = "F" THEN 5270
5220  IF F$ = "R" THEN 5240
5230  GOTO 5210
5240  HOME :S =  INT ( RND (1) * 10) + 1: IF S < 3 THEN 5260
5250  PRINT "THE SPIDER SHOOTS A HUGE WEB AT THE ": PRINT : PRINT "PARTY AND YOU
 A
     RE CAPTURED TO BE EATEN": PRINT : PRINT "BY SHELOB.": VTAB 22: PRINT "PRESS
 A
     NY KEY TO CONTINUE";: GET A$: GOTO 9000
5260  PRINT : PRINT "YOU RUN DOWN THE TUNNEL,TO THE EXIT.": VTAB 22: PRINT "PRES
S
     ANY KEY TO CONTINUE";: GET A$: GOTO 6000
5270  HOME : PRINT "ARAGORN AND YOU ATTACK SHELOB.": PRINT
5272 M = M + 500
5275 C =  INT ( RND (1) * 10) + 1:S =  INT ( RND (1) * 10) + 1
5277  IF C < 3 THEN 5290
5280  GOTO 5300
5290  PRINT "GANDALF SHOUTS 'ANNON EDHELLEN' AND ": PRINT : PRINT "SHELOB SHRIVE
LS
      AND DIES.": VTAB 22: PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 6000
5300  PRINT : PRINT "YOUR ";W$;" DOES TREMENDOUS DAMAGE TO": PRINT : PRINT "SHEL
OB
     'S EYES.FINALLY YOU SHATTER IT'S": PRINT : PRINT "HEAD.": VTAB 22: PRINT "P
RE
     SS ANY KEY TO CONTINUE";' GET A$
6000  HOME : PRINT "ONCE IN MORDOR,YOU QUICKLY FIND THE": PRINT : PRINT "CRACK O
F
     DOOM.STANDING BEFORE IT IS": PRINT : PRINT "SAURON,HIMSELF.THE PARTY IS PAR
AL
     IZED": PRINT : PRINT "BY FEAR."
6010  PRINT : PRINT "WILL YOU FIGHT OR SURRENDER TO SAURON ?";: GET F$
6011  PRINT
6015  IF F$ = "F" THEN 6100
6020  IF F$ = "S" THEN 6050
6025  GOTO 6010
6050  HOME : PRINT "YOUR PARTY IS TAKEN PRISONER AND THE": PRINT : PRINT "RULING
 R
     ING IS CONFISCATED."
6060  IF I$ = "RING" THEN 6065
```



LORD OF THE RINGS

```
6062  GOTO 6070
6065  PRINT : PRINT "THE DWARF RING FROM MORIA IS ALSO": PRINT : PRINT "TAKEN."
6070  PRINT : PRINT "BEFORE YOU ARE EXECUTED YOU ARE FORCED": PRINT : PRINT "TO
SE
     E SAURON ARISE TO CONQUER": PRINT : PRINT "MIDDLE-EARTH.": PRINT : PRINT :
PRINT
     "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 9000
6100  HOME : IF I$ = "RING" THEN 6150
6110 M = M + 5000
6120  GOTO 6200
6150  PRINT "YOU REMEMBER THE RING YOU FOUND IN ": PRINT : PRINT "MORIA.YOU SHOU
T
     IN A STRANGE TONGUE": PRINT : PRINT "'NAUR AN EDDAITH AMMEN!'"
6160 I =  INT ( RND (1) * 10) + 1
6165  IF I < 5 THEN 6169
6167  PRINT : PRINT "SAURON LAUGHS AT YOU AND BLASTS YOU ": PRINT : PRINT "WITH
LI
     GHTNING BOLTS.": PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
: GOTO
     9000
6169 M = M + 1000
6170  PRINT : PRINT "AS YOU SAY THIS SAURON WITHERS AND ": PRINT : PRINT "DIES.Y
OU
      ARE FREE TO DESTROY THE RING.": PRINT : PRINT : PRINT "PRESS ANY KEY TO CO
NT
     INUE";: GET A$: GOTO 7000
6200  PRINT "OVERCOMING YOUR FEAR,YOU AND ARAGORN": PRINT : PRINT "ATTACK SAURON
.H
     E RETALIATES WITH BOLTS": PRINT : PRINT "OF LIGHTNING."
6210  IF GA$ = "DEAD" THEN 6230
6220  PRINT : PRINT "GANDALF TRIES MANY SPELLS TO DESTROY": PRINT : PRINT "SAURO
N,
     BUT NONE SEEM TO WORK."
6230 S =  INT ( RND (1) * 15) + 1: IF GA$ = "DEAD" THEN 6240
6235  IF S < 5 THEN 6259
6236 M = M + 1000
6237  GOTO 6250
6240  IF S < 3 THEN 6259
6241 M = M + 2000
6242  GOTO 6250
6250  GOTO 6167
6259  PRINT
6260  IF GA$ = "DEAD" THEN 6270
6265  PRINT "THE COMBINED ATTACK OF ARAGORN,YOU AND ": PRINT : PRINT "GANDALF IS
 T
     OO MUCH FOR SAURON.HE IS": PRINT : PRINT "DEAD.YOU CAN NOW DESTROY THE RING
."
     : GOTO 6300
```

```
6266  RETURN
6270  PRINT "TOGETHER YOU AND ARAGORN DEFEAT THE ": PRINT : PRINT "DARK LORD.WEL
L
     DONE.YOU CAN NOW DESTROY": PRINT : PRINT "THE RING."
6300  PRINT : PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 7
00
     0
7000  HOME : PRINT "FRODO GIVES YOU THE RING TO DESTROY"
7010  PRINT : PRINT "WILL YOU DO IT ? ";: GET O$
7020  IF O$ = "N" THEN 7099
7025 O =  INT ( RND (1) * 10) + 1: IF O < 3 THEN 7090
7027 M = M + 1000
7030  PRINT : PRINT "YOU GAZE INTO THE CRACK OF DOOM AND SEE": PRINT : PRINT "GR
EE
     N FLAMES AND LAVA WITHIN.YOU TOSS": PRINT : PRINT "THE RULING RING IN AND W
AT
     CH IT MELT."
7040  IF I$ = "RING" THEN 7050
7045  GOTO 7300
7050  PRINT : PRINT "AS THE RULING RING MELTS,THE DWARF-RING": PRINT : PRINT "FR
OM
      MORIA GLOWS AND DISAPPEARS.": GOTO 7500
7090  HOME : PRINT "YOU FIND YOURSELF UNABLE TO PART WITH": PRINT : PRINT "THE R
IN
     G."
7095 E$ = "Y"
7099  PRINT
7100  PRINT : PRINT "YOU TAKE OVER THE FORCES OF MORDOR AND": PRINT : PRINT "PRO
CL
     AIM YOURSELF THE NEW DARK LORD": PRINT : PRINT : PRINT "HAIL ";N$;",LORD OF
 E
     VIL.": PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$: GOTO 760
0
7300  PRINT
7500  PRINT : PRINT : PRINT "PRESS ANY KEY TO CONTINUE";: GET A$
7510  HOME : PRINT "WELL DONE,";N$;"."
7520  PRINT : PRINT : PRINT "YOU ARE A CREDIT TO ";RS$;" EVERYWHERE."
7530  PRINT : PRINT : PRINT "AS A REWARD,ARAGORN (WHO IS NOW KING OF": PRINT : P
RINT
     "GONDOR) GIVES YOU ";M;" GOLD": PRINT : PRINT "PIECES FOR YOUR CONTINUED BR
AV
     ERY."
7600  PRINT : PRINT : PRINT : PRINT : PRINT "DO YOU WANT TO PLAY AGAIN ? ";: GET
 A
     $
7610  IF A$ = "Y" THEN 5
7620  HOME : SPEED= 255: END
```

■

RAIDERS

The object is to enter a maze of caves, acquire the golden idol and return.
Along the way you may pick up objects which will help you. For example the rope must be used to swing across the crevices and the shield is protection from the darts. Gold is used to buy maps. Torches and elixers fend off monsters.

On the map black indicates a wall and cannot be passed.

When darts are being fired at you, defend yourself using Paddle (O) to move the shield. You must knock out 4 of the 7 darts. When in battle with a monster use the key 'p' to bash him and 's' to shield...

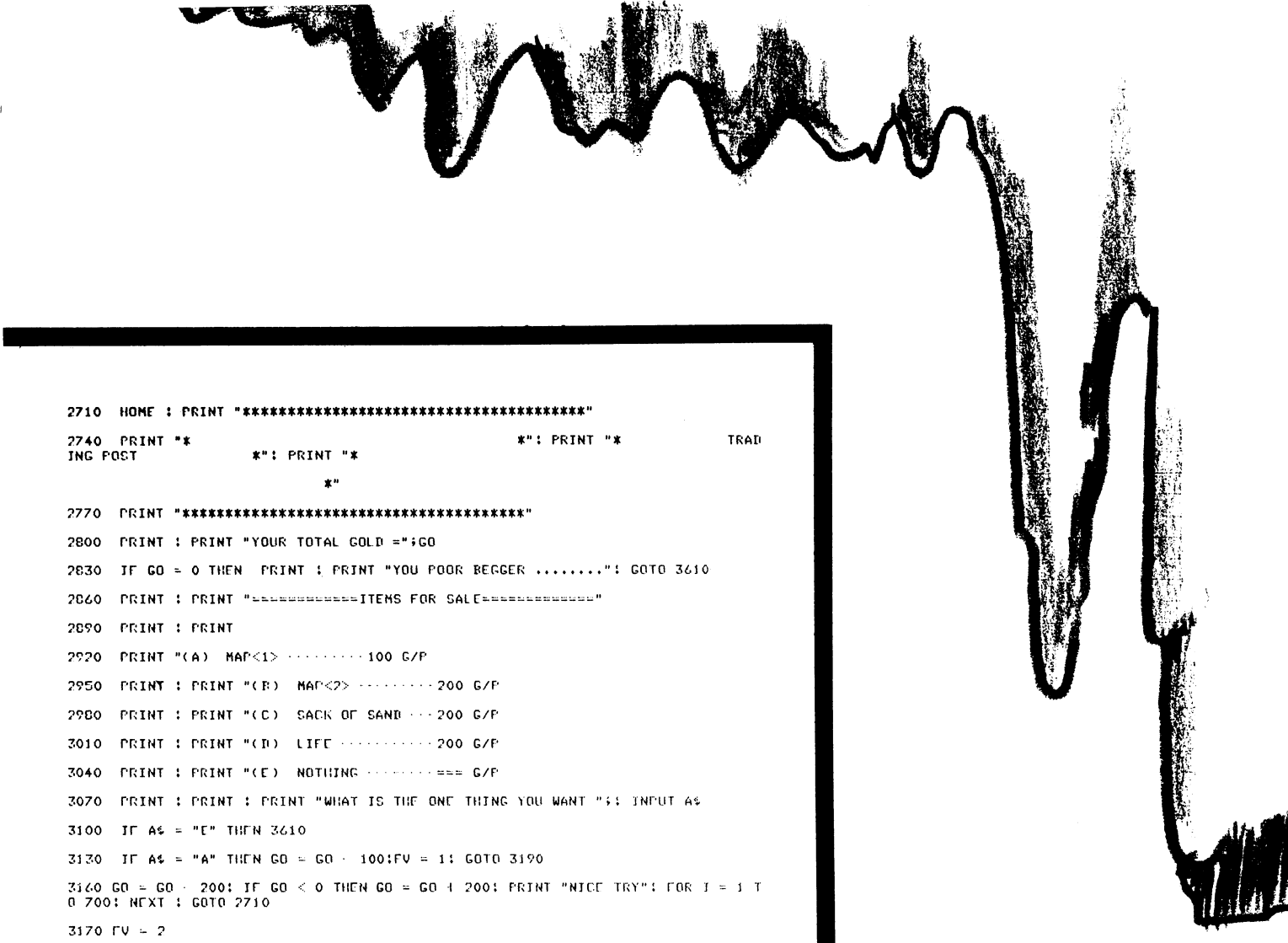
Mike Bantick
Mount Beauty Vic

```
10  DIM D(40,40)

100  DIM C$(13)

130 X = 20:Y = 38

160  HOME : INVERSE : PRINT "                                                  ": VTAB 5:
PRINT "
     "

170 LF = 10

190  FOR I = 2 TO 4: VTAB I: HTAB 1: PRINT " ": VTAB I: HTAB 39: PRINT " "

220  NEXT

250  NORMAL : VTAB 3: HTAB 2: PRINT "              EXPLORERS              "

310  VTAB 7: PRINT " THIS WILL TAKE APPROX'LY 55 SEC......."

340  VTAB 10: PRINT "         CREATING            MAZE

370 V = 20:S = 38

380  IF PS = 1 THEN V = 20:S = 1:X = 20:Y = 1

400 D(V,S) = 1:Ty =  INT (3 *  RND (1)) + 1: IF TY = 1 AND PS = 0 THEN S = S +
1: GOTO 490

420  IF PS = 1 AND TY = 1 THEN S = S + 1

430  IF TY = 2 THEN V = V - 1

460  IF TY = 3 THEN V = V + 1

490  IF S = 0 THEN 610

500  IF S = 39 THEN 610

520  IF V = 0 THEN V = 1

550  IF V = 39 THEN V = 38

580  GOTO 400

610  FOR I = 1 TO 400:V =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1
:D(V,S) = 1

640  VTAB 15: PRINT  TAB( I / 2)"."

670  NEXT

700  REM  PLACE OBJECTS.....

730  FOR I = 1 TO 100

760 D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1: IF D(D,S) = 2 TH
EN 760

790 D(D,S) = 2: NEXT

820  FOR I = 1 TO 70

850 D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1: IF D(D,S) > 1 TH
EN 850

880 D(D,S) = 3: NEXT

910  FOR I = 1 TO 50

940 D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1: IF D(D,S) > 1 TH
EN 940

970 D(D,S) = 4: NEXT
```

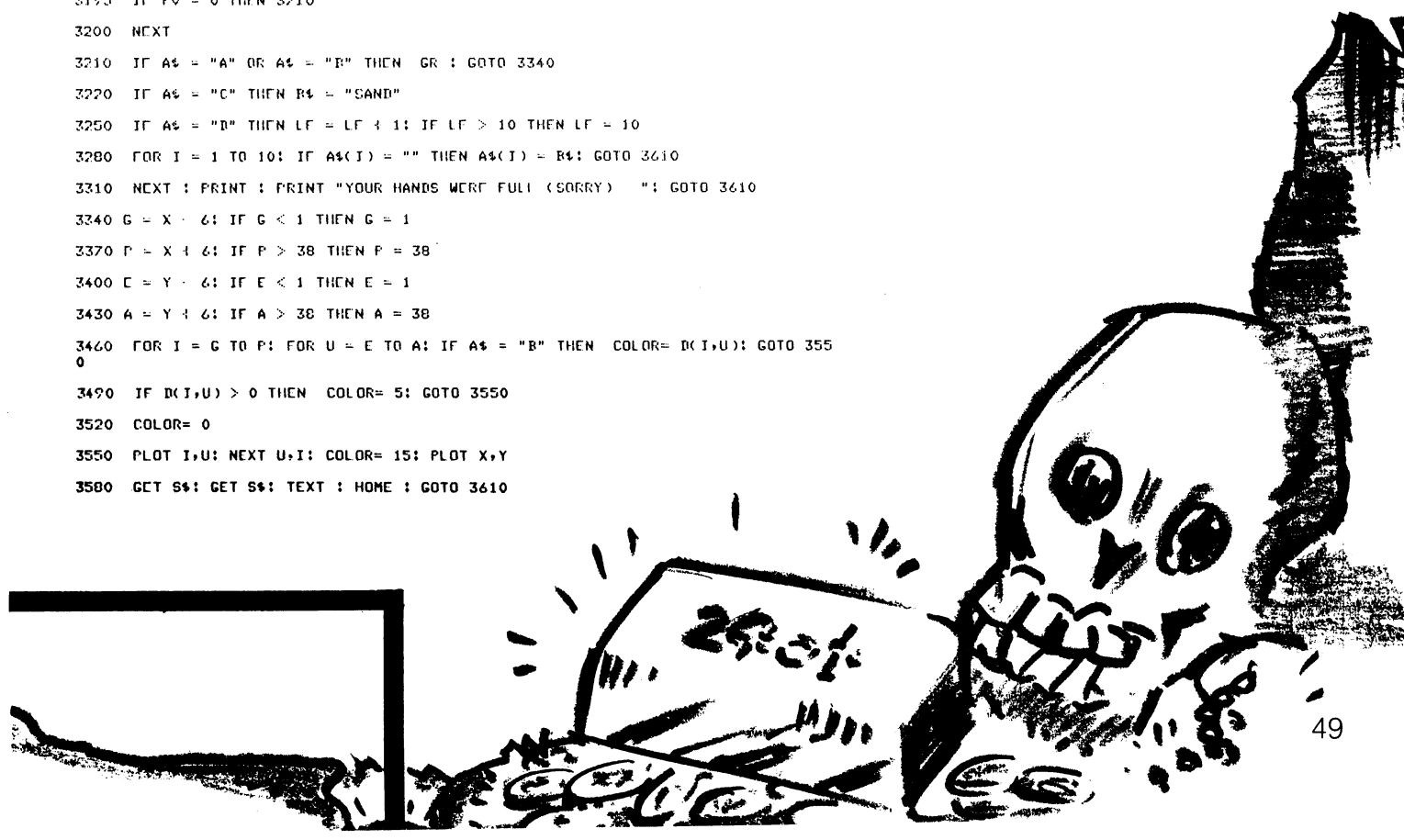
```
1000  FOR I = 1 TO 40

1030 D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1: IF D(D,S) > 1 T
HEN 1030

1052 D(I,U) = 0

1060 D(D,S) = 5: NEXT

1090  FOR I = 1 TO 10

1120 D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1: IF D(D,S) > 1 T
HEN 1120

1150 D(D,S) = 6: NEXT

1180  FOR I = 1 TO 3

1210 D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) + 1: IF D(D,S) > 1 T
HEN 1210

1240 D(D,S) = 7: NEXT

1270  REM  52 SECONDS

1300  FOR I = 1 TO 200:D =  INT (38 *  RND (1)) + 1:S =  INT (38 *  RND (1)) +
1:F =  INT (6 *  RND (1)) + 8

1330 D(D,S) = F: NEXT

1360  FOR I = 2 TO 13: READ C$(I): NEXT

1390  DATA  GOLD,SHIELD,TORCH,ROPE,ELIXER,SAND,TRADER,DARTS,TRONS,WUP,LOPTROP,C
REVICE

1420  HOME : ON D(X,Y) GOTO 1450,1930,1930,1930,1930,1930,1930,2710,7000,3670,3
670,3670,5440

1430  RESTORE

1450  PRINT : PRINT : PRINT "YOU CAN MOVE .....        LIVES ";LF

1460  IF LF <  = 0 THEN  END

1480  NORMAL

1500 PF = 0

1510  IF D(X + 1,Y) > 0 THEN  PRINT : PRINT "EAST":PF = PF + 1

1540  IF D(X - 1,Y) > 0 THEN  PRINT : PRINT "WEST":PF = PF + 1

1570  IF D(X,Y - 1) > 0 THEN  PRINT : PRINT "NORTH":PF = PF + 1

1600  IF D(X,Y + 1) > 0 THEN  PRINT : PRINT "SOUTH":PF = PF + 1

1610  IF PS = 1 AND PF = 1 AND SP = 1 THEN  PRINT : PRINT "OH ! NO! A DEAD END
 /////SQUISH///////":LF = LF -  INT (5 *  RND
     (1)) + 1:SP = 0

1630  PRINT : PRINT : PRINT "WHICH 'N,E,W,S' ";: INPUT A$: IF A$ <  > "N" AND A
$ <  > "S" AND A$ <  > "E" AND A$ <  > "W" THEN
      PRINT : PRINT "YOU MUST MOVE": GOTO 1450

1660  IF A$ = "N" AND D(X,Y - 1) = 0 THEN  PRINT : PRINT "OUCH": GOTO 1450

1690  IF A$ = "S" AND D(X,Y + 1) = 0 THEN  PRINT : PRINT "OUCH": GOTO 1450
```

RAIDERS

```
1720  IF A$ = "E" AND D(X + 1,Y) = 0 THEN  PRINT : PRINT "OUCH": GOTO 1450

1750  IF A$ = "W" AND D(X - 1,Y) = 0 THEN  PRINT : PRINT "OUCH": GOTO 1450

1780  IF A$ = "N" THEN Y = Y - 1: IF Y = 1 THEN 10000

1810  IF A$ = "E" THEN X = X + 1

1840  IF A$ = "W" THEN X = X - 1

1870  IF A$ = "S" THEN Y = Y + 1: IF PS = 1 AND Y = 38 THEN 12000

1900  GOTO 1420

1930  HOME : PRINT : PRINT "THERE IS A ";C$(D(X,Y));". DO YOU

1960  PRINT : PRINT "PICK IT UP (P) OR LEAVE IT (L) ";: INPUT A$: IF A$ < > "L
" AND A$ < > "P" THEN 1930

1990  IF A$ = "L" THEN  PRINT : PRINT "OK !!     ";C$(D(X,Y));" LEFT  ": FOR I =
 1 TO 1000: NEXT : GOTO 1450

2020 D$ = C$(D(X,Y))

2050  PRINT : PRINT "OK !!  ";D$;" PICKED UP

2080  FOR I = 1 TO 10: IF A$(I) = "" THEN A$(I) = D$: GOTO 2140

2110  NEXT I: PRINT : PRINT "YOUR HANDS ARE FULL": GOTO 1960

2140 D(X,Y) = 1: IF D$ = "GOLD" THEN GO = GO + 100

2170 WE = 0:GO = 0: FOR I = 1 TO 10: IF A$(I) = "" THEN 2270

2180 D$ = A$(I): IF D$ = "GOLD" OR D$ = "SHIELD" OR D$ = "SAND" THEN WE = WE +
100

2185  IF D$ = "GOLD" THEN GO = GO + 100

2190  IF D$ = "TORCH" THEN WE = WE + 50

2200  IF D$ = "ROPE" THEN WE = WE + 30

2210  IF D$ = "ELIXER" THEN WE = WE + 10

2270  NEXT

2290  FOR I = 1 TO 1500: NEXT : HOME : PRINT "            INVENTORY.........


2320  PRINT : PRINT

2350  FOR I = 1 TO 10: IF A$(I) = "" THEN 2410

2380  PRINT : PRINT A$(I)

2410  NEXT

2440  IF WE > 500 THEN 2500

2470  FOR I = 1 TO 2000: NEXT : GOTO 1420

2500  PRINT : PRINT "TO HEAVY..WHICH DO YOU WANT TO DROP ";: INPUT A$: FOR I =
1 TO 10: IF A$ < > A$(I) THEN  NEXT I: PRINT

     : PRINT "YOU DONT HAVE ";A$: GOTO 2290

2530  IF A$ = "GOLD" THEN GO = GO - 100

2660 A$(I) = ""

2680  GOTO 2170
```

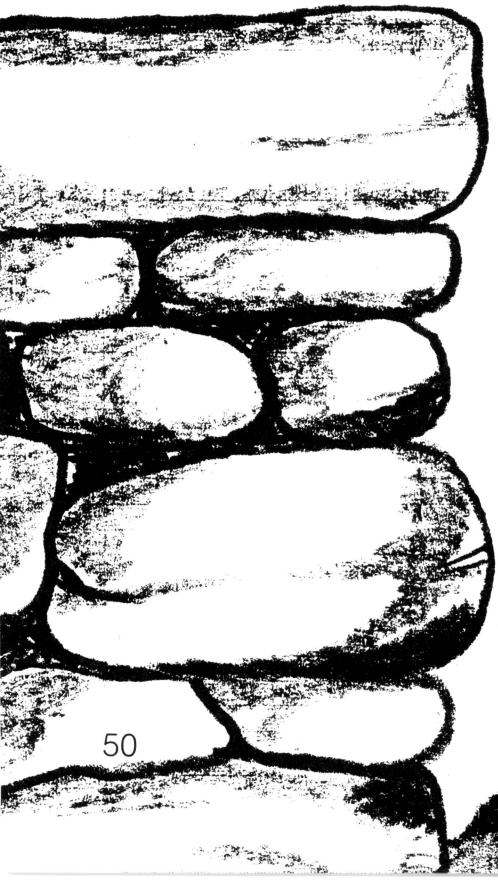


```
2710  HOME : PRINT "****************************************"

2740  PRINT "*                                    *": PRINT "*            TRAD
ING POST            *": PRINT "*

                                *"

2770  PRINT "****************************************"

2800  PRINT : PRINT "YOUR TOTAL GOLD =";GO

2830  IF GO = 0 THEN  PRINT : PRINT "YOU POOR BEGGER ........": GOTO 3610

2860  PRINT : PRINT "============ITEMS FOR SALE============="

2890  PRINT : PRINT

2920  PRINT "(A)  MAP<1> ..........100 G/P

2950  PRINT : PRINT "(B)  MAP<2> ..........200 G/P

2980  PRINT : PRINT "(C)  SACK OF SAND ...200 G/P

3010  PRINT : PRINT "(D)  LIFE ............200 G/P

3040  PRINT : PRINT "(E)  NOTHING ........=== G/P

3070  PRINT : PRINT : PRINT "WHAT IS THE ONE THING YOU WANT ";: INPUT A$

3100  IF A$ = "E" THEN 3610

3130  IF A$ = "A" THEN GO = GO - 100:FV = 1: GOTO 3190

3160 GO = GO - 200: IF GO < 0 THEN GO = GO + 200: PRINT "NICE TRY": FOR I = 1 T
O 700: NEXT : GOTO 2710

3170 FV = 2

3190  FOR I = 1 TO 10: IF A$(I) = "GOLD" THEN A$(I) = "":FV = FV - 1

3195  IF FV = 0 THEN 3210

3200  NEXT

3210  IF A$ = "A" OR A$ = "B" THEN  GR : GOTO 3340

3220  IF A$ = "C" THEN B$ = "SAND"

3250  IF A$ = "D" THEN LF = LF + 1: IF LF > 10 THEN LF = 10

3280  FOR I = 1 TO 10: IF A$(I) = "" THEN A$(I) = B$: GOTO 3610

3310  NEXT : PRINT : PRINT "YOUR HANDS WERE FULL (SORRY)   ": GOTO 3610

3340 G = X - 6: IF G < 1 THEN G = 1

3370 P = X + 6: IF P > 38 THEN P = 38

3400 E = Y - 6: IF E < 1 THEN E = 1

3430 A = Y + 6: IF A > 38 THEN A = 38

3460  FOR I = G TO P: FOR U = E TO A: IF A$ = "B" THEN  COLOR= D(I,U): GOTO 355
0

3490  IF D(I,U) > 0 THEN  COLOR= 5: GOTO 3550

3520  COLOR= 0

3550  PLOT I,U: NEXT U,I: COLOR= 15: PLOT X,Y

3580  GET S$: GET S$: TEXT : HOME : GOTO 3610
```

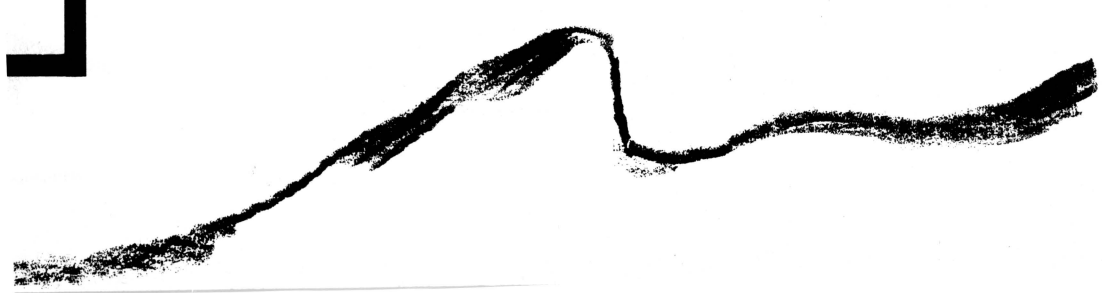


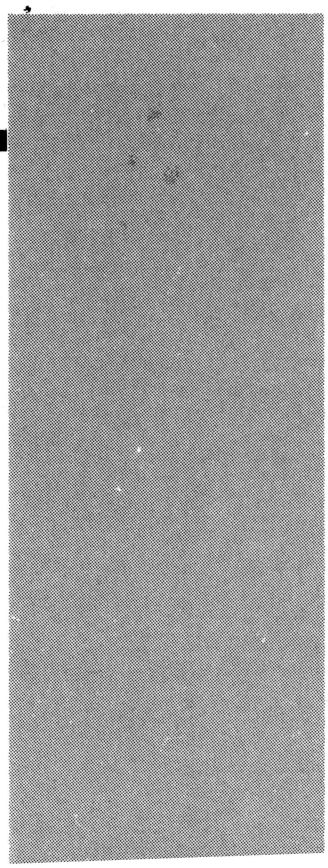
RAIDERS

```
4420  FOR I = 1 TO 200: NEXT

4450  GOTO 4270

4480  VTAB 21: PRINT "                ENGAGE...               "

4510 ZF =  INT (3 *  RND (1)) + 1: ON ZF GOTO 4630,4930,5020

4540  IF SM < 1 THEN  HOME : PRINT "     YOU HAVE DEFEATED THE ";D$: FOR I = 1
TO 2000: NEXT :D(X,Y) = 1: TEXT : HOME : GOTO

     1420

4570  IF SY < 1 THEN  HOME : PRINT "   YOU HAVE BEEN DEFEATED 'BAD LUCK!'":LF =
 LF   1: GOTO 5290

4600  GOTO 5020

4630  COLOR= 0: FOR I = 11 TO 20: HLIN 21,26 AT I: NEXT I

4660 SD = SD + 1

4690  IF SD = 2 THEN 4780

4720  COLOR= 8: HLIN 20,21 AT 20: VLIN 15,19 AT 21: VLIN 13,15 AT 22: VLIN 11,1
2 AT 23: VLIN 11,12 AT 24

4750  GOTO 4810

4780  COLOR= 8: HLIN 20,21 AT 20: VLIN 18,19 AT 22: VLIN 16,17 AT 23: VLIN 14,1
5 AT 24: VLIN 14,15 AT 25: VLIN 12,13 AT 25: VLIN

     12,13 AT 26

4810  IF SD = 2 THEN SD = 0: GOTO 4870

4840  FOR I = 1 TO 200: NEXT : GOTO 4540

4870  IF CV <  > 2 THEN SY = SY - 1: FOR I = 1 TO 10:KC =  PEEK (49200) *  PEEK
 (   16336): NEXT

4900  GOTO 4540

4930  COLOR= 0: FOR I = 11 TO 20: HLIN 21,26 AT I: NEXT I

4960  COLOR= 9: HLIN 20,22 AT 20: VLIN 13,19 AT 22

4990  GOTO 4540

5020 ZZ =  PEEK ( - 16384): IF ZZ = 208 THEN CV = 1

5050  IF ZZ = 211 THEN CV = 2

5080  IF CV = 1 THEN 5170

5110  IF CV <  > 2 THEN CV = 0: GOTO 4510

5140  COLOR= 2: HLIN 26,27 AT 19: VLIN 13,18 AT 26: GOTO 4510

5170  COLOR= 11: PLOT 27,18: HLIN 25,26 AT 17: PLOT 25,17: VLIN 14,17 AT 24: VL
IN 12,15 AT 23: VLIN 12,13 AT 22

5200  IF ZF = 1 THEN  FOR I = 1 TO 10:GH =  PEEK ( - 16336) +  PEEK ( - 16336)
   PEEK ( - 49200): NEXT :SM = SM - 1

5230  POKE  - 16368,0:CV = 0
```



```
3610  FOR I = 1 TO 2000: NEXT :D(X,Y) = 1: GOTO 1420

3670  HOME : PRINT "                    MONSTERS          "

3700  PRINT "                    ()()()()

3730  PRINT : PRINT : PRINT : PRINT " PREPARE YOURSELF AND USE....

3760  PRINT : PRINT : PRINT "  (S) FOR SHIELD

3790  PRINT : PRINT "  (P) TO STRIKE

3820  PRINT : PRINT "DO YOU WANT TO USE AN ELIXER,OR TORCH  "

3850  PRINT : PRINT "TO DEFEND YOURSELF (E,T OR N 'NONE') ";: INPUT A$:SM =  IN
T (14 *  RND (1)) + 3:SY =  INT (8 *  RND (1))

      + 3

3880 D$ = C$(D(X,Y))

3910  IF A$ = "N" THEN 4150

3940  IF A$ = "E" THEN 4030

3970  FOR I = 1 TO 10: IF A$(I) <  > "TORCH" THEN  NEXT I: PRINT "YOU HAVN'T GO
T A TORCH": GOTO 3820

3980 A$(I) = ""

4000  GOTO 4060

4030  FOR I = 1 TO 10: IF A$(I) <  > "ELIXER" THEN  NEXT I: PRINT "YOU HAVN'T G
OT AN ELIXER": GOTO 3820

4040 A$(I) = ""

4060  IF D$ = "WUP" AND A$ = "T" THEN  PRINT : PRINT "TORCH LOWERS WUP'S CONFID
ENCE  ":SM = SM - 2

4090  IF D$ = "TRONS" AND A$ = "E" THEN  PRINT : PRINT "ELIXER AFFECTS TRONS  "
:SM = SM - 2

4120  FOR I = 1 TO 2000: NEXT

4150  GR : COLOR= 1: VLIN 12,27 AT 30: VLIN 12,24 AT 29

4180  VLIN 18,22 AT 28: VLIN 12,15 AT 31: VLIN 12,15 AT 28: HLIN 28,29 AT 27: C
OLOR= 15: PLOT 28,13

4210  HLIN 27,29 AT 20: PLOT 29,19

4240 S = 10

4270 : COLOR= 0: FOR I = 15 TO 27: HLIN S - 10,S + 5 AT I: NEXT :S = S + 1

4300  COLOR= 6: HLIN S - 10,S - 3 AT 26: HLIN S - 7,S - 2 AT 25: HLIN S - 3,S +
 2 AT 24: VLIN 20,23 AT S - 2: VLIN 19,23 AT S

      - 1: VLIN 18,27 AT S

4330  VLIN 17,27 AT S + 1: VLIN 17,23 AT S + 2: VLIN 15,23 AT S + 3: VLIN 15,23
 AT S + 4: VLIN 15,18 AT S + 5: HLIN S + 2,S +

     4 AT 27: COLOR= 15: PLOT S + 5,16

4360  VLIN 21,23 AT S: PLOT S + 1,21

4390  IF S = 15 THEN 4480
```





RAIDERS

```
5260  GOTO 4510
5290  FOR I = 1 TO 10: IF A$(I) <  > "" THEN 5350
5320  NEXT : PRINT : PRINT "YOU HAVE NOTHING WORTH STEALING     ": GOTO 5380
5350  PRINT : PRINT "THE ";D$;" HAS STOLEN YOUR ";A$(I): IF A$(I) = "GOLD" THEN
 GO = GO - 100
5360 A$(I) = ""
5380  FOR I = 1 TO 2000: NEXT : TEXT : HOME
5410 D(X,Y) = 1: GOTO 1420
5440  FOR I = 1 TO 10: IF A$(I) = "ROPE" THEN A$(I) = "": GOTO 6130
5470  NEXT
5500  HOME : PRINT "                    CREVICE
5530  PRINT : PRINT : PRINT "   OH !! NO !!  .....ARGHHHHHHHHHHHHHH": FOR I = 1
 TO 25: NF =  PEEK ( - 16336) +  PEEK ( - 16336
     ) +  PEEK (49200) -  PEEK ( - 16336): NEXT
5040 LF = LF - 1
5560  FOR I = 1 TO 1000: NEXT : GR : COLOR= 8
5590  FOR I = 20 TO 39: HLIN 0, INT (5 *  RND (1)) + 17 AT I: HLIN  INT (5 *  R
ND (1)) + 23,39 AT I: NEXT
5620 XX = 5:YY = 15
5650  COLOR= 0: FOR I = YY - 5 TO YY + 3: HLIN XX - 4,XX + 4 AT I: NEXT
5680  COLOR= 7: HLIN XX - 3,XX - 2 AT YY - 4: PLOT XX - 2,YY - 3: HLIN XX - 1,X
X + 2 AT YY - 2: HLIN XX,XX + 1 AT YY - 1: HLIN
     XX,XX + 1 AT YY
5710  PLOT XX + 3,YY - 1: HLIN XX + 3,XX + 4 AT YY
5740  COLOR= 2: PLOT XX,YY - 4: COLOR= 10
5770  HLIN XX,XX + 3 AT YY + 1: PLOT XX + 3,YY + 2: HLIN XX - 1,XX AT YY + 3: P
LOT XX,YY + 2
5800  IF XX = 20 THEN YY = YY + 1: PRINT "ARGHHHHHHHHHH": GOTO 5860
5830 XX = XX + 1
5860  IF YY = 35 THEN 5920
5890  GOTO 5650
5920  TEXT : HOME : VTAB 10: PRINT "            SPLATTTTT !!!!            ": F
OR I = 1 TO 1000: NEXT I
5950  FOR I = 1 TO 10:DF =  INT (10 *  RND (1)) + 1: IF A$(DF) <  > "" THEN 601
0
5980  NEXT : GOTO 6280
```

```
6010  PRINT : PRINT : PRINT "YOU HAVE LOST YOUR "A$(DF)

6040  IF A$(DF) = "GOLD" THEN GO = GO - 100

6070 A$(DF) = "": FOR I = 1 TO 2000: NEXT

6100  GOTO 6250

6130  HOME : PRINT : PRINT "  OH !! NO ... A CRAVICE

6160  PRINT : PRINT "LUCKY YOU HAD THAT PIECE OF ROPE TO"

6190  PRINT : PRINT "SWING ACROSS WITH ....."

6220 A$(I) = ""

6250  FOR I = 1 TO 3500: NEXT

6280 D(X,Y) = 1: GOTO 1420

7000  REM      DARTS

7010  HOME : PRINT : PRINT "<<<<<<<<<<<<<POISON DARTS>>>>>>>>>>>>>"

7020  PRINT : PRINT : FOR I = 1 TO 10: IF  A$(I) <  > "SHIELD" THEN  NEXT : PRIN
T : PRINT "OH DEAR,, YOU HAVEN'T GOT A SHIELD

        ": GOTO 7500

7025 A$(I) = ""

7030  GR : COLOR= 4: FOR I = 0 TO 39: HLIN 0,39 AT I: NEXT : COLOR= 6: FOR I =
18 TO 22: VLIN 34,37 AT I: NEXT

7035  COLOR= 9: PLOT 20,33: COLOR= 0: PLOT 23,35: PLOT 17,35

7040 S = 20: FOR I = 1 TO 7:C(I) =  INT (10 *  RND (1)) + 1:P(I) =  INT (20 *
RND (1)) + 10: NEXT

7045 PF = 0

7046 ST = 0

7050  COLOR= 4: HLIN S - 1,S + 1 AT 30:GF =  PDL (0) / 8 + 4: IF GF < S THEN S
= S - 1

7060  IF GF > S THEN S = S + 1

7070  COLOR= 13: HLIN S - 1,S + 1 AT 30

7080  FOR I = 1 TO 7: COLOR= 4: PLOT P(I),C(I):C(I) = C(I) + 1: IF C(I) = 35 TH
EN  PRINT "               OUCH !!

        ":PF = PF + 1:ST = ST + 1:P(I) = 0:C(I) = 0

7090  COLOR= 15: PLOT P(I),C(I): IF  SCRN( P(I),C(I) + 1) = 13 THEN P(I) = 0:C(
I) = 0:ST = ST + 1

7100  IF ST = 7 THEN 7400

7110  NEXT I: GOTO 7050

7400  IF PF > 3 THEN  PRINT : PRINT "    OH !! DEAR THAT HURT.............": 
GOTO 7500

7410  PRINT : PRINT "    ////////SAVED///////  ": FOR I = 1 TO 3000: NEXT : TE
```

RAIDERS

▽

```
XT : HOME :D(X,Y) = 1

7420  GOTO 1450

7500  FOR I = 1 TO 10:DF =  INT (10 *  RND (1)) + 1: IF A$(DF) <  > "" THEN 752
0

7510  NEXT : PRINT : PRINT "NOTHING WAS DAMAGED.....": FOR I = 1 TO 1500: NEXT
: GOTO 7410

7520  PRINT : PRINT "THE "A$(DF)" WAS DAMAGED BEYOND REPAIR"

7530 D$ = A$(DF):A$(DF) = ""

7540  IF D$ = "GOLD" THEN GO = GO - 100

7550 LF = LF - 1

7590  FOR I = 1 TO 3500: NEXT : TEXT : HOME :D(X,Y) = 1: GOTO 1450

10000  RESTORE : HOME : PRINT : PRINT "CONGRATULATIONS YOU ARE HALF WAY......."

10010  FOR I = 1 TO 10: IF A$(I) = "SAND" THEN 10100

10020  NEXT : PRINT : PRINT "OH ! GOD YOU DIDN'T HAVE A SACK OF SAND"

10030  PRINT : PRINT "TO PUT IN PLACE OF THE IDOL..YOU CAN

10040  PRINT : PRINT "HEAR A RUMBLE AND A ENORMOUS STONE

10050  PRINT : PRINT "SPHERE IS ROLLING DOWN AT YOU...DO NOT "

10060  PRINT : PRINT "RUN INTO A DEAD END OR YOU WILL BE "

10070  PRINT : PRINT "SQUISHED......

10075 SP = 1

10080  GOTO 10150

10100  PRINT : PRINT "YOU NOW HAVE THE GOLDEN IDOL SO TRY TO "

10120  PRINT : PRINT "TRY TO MAKE IT BACK TO THE ENTRANCE

10150 PS = 1

10151  FOR I = 1 TO 38: FOR U = 1 TO 38: IF D(I,U) > 8 THEN 10155

10152 D(I,U) = 0

10155  NEXT U,I

10160  GOTO 380

12000  HOME : PRINT : PRINT "WELL DOWN ========

12010  PRINT : PRINT "YOU HAVE SURVIVED///////
```

■

TIME PILOT

This is an action/low resolution program for the Apple II. You control an aircraft able to travel through time zones and encounter a variety of enemies in each zone.

Starting in the year 1910 you battle the bi-planes that zoom from all directions. Use the arrow keys to rotate the jet 45 degrees in any direction. Pressing the space bar fires missiles (missiles? in 1910? Ed) unless there are already two missiles on the screen.

Each time an enemy aircraft passes over your central jet you lose a certain amount of shielding depending on how long the enemy stays there. As each enemy craft is shot down the red line at the top of the screen recedes until you have amassed a total of 40 hits. Large alien craft appear at the top of the screen. When destroyed they are worth 500 points and transfers your jet to the next time zone.

Mike Bantick
Mount Beauty Vic

APPLE II

```
]LIST

5  REM      TIME PILOT
6  HOME
10  FOR I = 1 TO 23: READ N: POKE
    769 + I,N: NEXT
12  DATA     173,48,192,136,208,5
    ,206,1,3,240,9,202,208,245,1
    74,0,3,76,2,3,96,0,0
50 L(1) = 5:L(2) = 2:L(3) = 1:L(4
    ) = 6:L(5) = 3
55 T(1) = 1910:T(2) = 1940:T(3) =
    1970:T(4) = 1983:T(5) = 2001

56 PL = 3
60 V = 1:LL = 15
65 SH = 40
100  TEXT : HOME
101  IF V > 5 THEN V = 1:LL = LL -
    5: IF LL < 5 THEN LL = 5
102  POKE  - 16304,0: POKE  - 163
    02,0: POKE  - 16300,0: POKE
     - 16298,0: COLOR= L(V) + 20

103 XX =  INT (30 *  RND (1)) + 5
    :YY =  INT (30 *  RND (1)) +
    5
104  FOR I = 0 TO 39: VLIN 0,47 AT
    I: NEXT I
105 BG = 0
107 X =  INT (30 *  RND (1)) + 5:
    Y =  INT (30 *  RND (1)) + 5
    :SY = 1:SX = 1
108  COLOR=  INT (20 *  RND (1)) +
    1: FOR I = 1 TO 350
110  PLOT X,Y:X = X + SX:Y = Y +
    SY: IF X > 38 THEN SX =  - 1

115  IF X < 1 THEN SX = 1
117  IF Y < 1 THEN SY = 1
118  IF Y > 46 THEN SY =  - 1
119  IF I / 3 =  INT (I / 3) THEN
     POKE 768,L(V) * 10: POKE 76
    9,7: CALL 770
120  NEXT I
140  FOR I = 1 TO 1500: NEXT I: TEXT
    : HOME : VTAB 10: HTAB 18: PRINT
    "YEAR": INVERSE : VTAB 12: HTAB
    18: PRINT T(V): NORMAL
145  FOR I = 1 TO 1500: NEXT I
155  GR
156  IF RS = 1 THEN RS = 0: GOTO
    170
160 NU = 39
170  COLOR= L(V) + 1: FOR I = 0 TO
    39: HLIN 0,39 AT I: NEXT I: COLOR
175 TT = L(V) + 1
180  FOR I = 1 TO 3:C(I) = 0:X(I)
     = 0: NEXT I

190 R = 4: GOSUB 1000
199  REM  START LOOP
200  FOR I = 1 TO 2: IF C(I) = 0 THEN
    240
210  COLOR= TT: PLOT C(I),Z(I):C(
    I) = C(I) + BX(I):Z(I) = Z(I
    ) + BY(I)
220  IF C(I) < 0 OR C(I) > 39 OR
    Z(I) < 1 OR Z(I) > 39 THEN C
    (I) = 0: GOTO 240
230  COLOR= 9: PLOT C(I),Z(I)
240  NEXT I
245  IF BG = 1 THEN  GOSUB 4000
246 TK = SH
250  FOR I = 1 TO 3: IF X(I) = 0 THE
     GOSUB 2000: GOTO 300
260  COLOR= TT: FOR U = Y(I) - 2 TO
    Y(I) + 2: HLIN X(I) - 3,X(I)
     + 3 AT U: NEXT U
265 X(I) = X(I) - SX + SX(I):Y(I)
     = Y(I) - SY + SY(I):SX(I) =
    SX(I) +  RND (1) - .5:SY(I) =
    SY(I) +  RND (1) - .5
270  IF X(I) < 3 OR X(I) > 35 OR
    Y(I) < 4 OR Y(I) > 35 THEN X
    (I) = 0: GOTO 300
280  GOSUB 3000
282  FOR U = 1 TO 2: IF C(U) > X(
    I) - 4 AND C(U) < X(I) + 4 AND
    Z(U) < Y(I) + 4 AND Z(U) > Y
    (I) - 4 THEN  GOSUB 2100: GOTO
    300
283  NEXT U
285  IF X(I) > 13 AND X(I) < 27 AND
    Y(I) > 13 AND Y(I) < 27 THEN
    KL =  PEEK ( - 16336) -  PEEK
    ( - 16336):SH = SH - 1
300  NEXT I
301  IF TK <  > SH THEN  GOSUB 10
    00
305 FF = R
310 Z =  PEEK ( - 16384): POKE  -
    16368,0: IF Z = 136 THEN R =
    R - 1: IF R < 1 THEN R = 8
315  IF Z = 149 THEN R = R + 1: IF
    R > 8 THEN R = 1
320  IF FF <  > R THEN  GOSUB 100
    0
330  IF Z = 160 THEN  POKE 768,50
    : POKE 769,20: CALL 770: GOTO
    340
335  GOTO 400
340  FOR I = 1 TO 2: IF C(I) = 0 TH
    BX(I) = 2 * SX:BY(I) = 2 * S
    Y: GOTO 345
342  NEXT I: GOTO 400
345  IF R = 1 THEN C(I) = 20:Z(I)
     = 15
346  IF R = 2 THEN C(I) = 24:Z(I)
     = 16
347  IF R = 3 THEN C(I) = 25:Z(I)
     = 20
348  IF R = 4 THEN C(I) = 24:Z(I)
     = 24
349  IF R = 5 THEN C(I) = 20:Z(I)
     = 25
350  IF R = 6 THEN C(I) = 16:Z(I)
     = 24
351  IF R = 7 THEN C(I) = 15:Z(I)
     = 20
352  IF R = 8 THEN C(I) = 16:Z(I)
     = 16
400  VTAB 21: PRINT "SCORE "SC"
      SHIELDING "SH" "
410  IF SH < 1 THEN  GOSUB 5000
800  GOTO 200
999  END
1000  COLOR= TT: FOR I = 16 TO 24
    : VLIN 16,24 AT I: NEXT I: ON
    R GOTO 1010,1040,1070,1100,1
    130,1160,1190,1220
1010  COLOR= 15: HLIN 16,24 AT 22
    : HLIN 16,24 AT 23: HLIN 17,
    23 AT 21: HLIN 18,22 AT 20: HLIN
    18,22 AT 19: HLIN 19,21 AT 1
    7: HLIN 19,21 AT 18
1011  HLIN 16,17 AT 24: HLIN 23,2
    4 AT 24
1015  COLOR= 9: PLOT 20,16: PLOT
    17,20: PLOT 23,20: HLIN 19,2
    1 AT 24: COLOR= 2: VLIN 18,1
    9 AT 20
1017 SX = 0:SY =  - 1
1020  RETURN
1040  COLOR= 15: HLIN 16,22 AT 20
    : HLIN 16,23 AT 19: HLIN 18,
    23 AT 18: HLIN 21,23 AT 17: HLIN
    19,22 AT 21: HLIN 20,22 AT 2
    2: HLIN 20,21 AT 23: HLIN 20
    ,21 AT 24
1042  COLOR= 9: PLOT 18,21: PLOT
    19,22: PLOT 23,20: PLOT 20,1
    7: PLOT 24,16: COLOR= 2: PLOT
    21,19: PLOT 22,18
1045 SX = .5:SY =  - .5
1050  RETURN
1070  COLOR= 15: VLIN 16,24 AT 17
    : VLIN 16,24 AT 18: VLIN 17,
    23 AT 19: VLIN 18,22 AT 20: VLIN
    18,22 AT 21: VLIN 19,21 AT 2
    3: VLIN 19,21 AT 22: VLIN 16
    ,17 AT 16: VLIN 23,24 AT 16:
     COLOR= 9
1072  PLOT 24,20: VLIN 19,21 AT 1
    6: PLOT 20,17: PLOT 20,23: COLOR=
    2: HLIN 21,22 AT 20
1075 SX = 1:SY = 0
1080  RETURN
1100  COLOR= 15: HLIN 16,22 AT 20
    : HLIN 16,23 AT 21: HLIN 18,

    23 AT 22: HLIN 21,23 AT 23: VLIN
    16,19 AT 20: VLIN 16,19 AT 2
    1: VLIN 18,19 AT 22: PLOT 19
    ,19: COLOR= 9
1102  PLOT 19,18: PLOT 18,19: PLOT
    24,24: PLOT 20,23: PLOT 23,2
    0: COLOR= 2: PLOT 21,21: PLOT
    22,22
1105 SX = .5:SY = .5
1110  RETURN
1130  COLOR= 15: HLIN 16,24 AT 17
    : HLIN 16,24 AT 18: HLIN 17,
    23 AT 19: HLIN 18,22 AT 20: HLIN
    18,22 AT 21: HLIN 19,21 AT 2
    2: HLIN 19,21 AT 23: HLIN 16
    ,17 AT 16: HLIN 23,24 AT 16:
     COLOR= 9
1132  HLIN 19,21 AT 16: PLOT 20,2
    4: PLOT 17,20: PLOT 23,20: COLOR=
    2: VLIN 21,22 AT 20
1135 SX = 0:SY = 1
1140  RETURN
1160  COLOR= 15: VLIN 16,22 AT 20
    : VLIN 16,23 AT 19: VLIN 18,
    23 AT 18: VLIN 21,23 AT 17: HLIN
    24,21 AT 20: HLIN 24,21 AT 2
    1: HLIN 22,21 AT 22: PLOT 21
    ,19: COLOR= 9
1162  PLOT 16,24: PLOT 21,18: PLOT
    22,19: PLOT 17,20: PLOT 20,2
    3: COLOR= 2: PLOT 18,22: PLOT
    19,21
1165 SX =  - .5:SY = .5
1170  RETURN
1190  COLOR= 15: VLIN 16,24 AT 23
    : VLIN 16,24 AT 22: VLIN 17,
    23 AT 21: VLIN 18,22 AT 20: VLIN
    18,22 AT 19: VLIN 19,21 AT 1
    8: VLIN 19,21 AT 17: VLIN 16
    ,17 AT 24: VLIN 23,24 AT 24:
     COLOR= 9
1192  VLIN 19,21 AT 24: PLOT 20,1
    7: PLOT 20,23: PLOT 16,20: COLOR=
    2: HLIN 18,19 AT 20
1195 SX =  - 1:SY = 0
1200  RETURN
1220  COLOR= 15: HLIN 18,24 AT 20
    : HLIN 17,24 AT 19: HLIN 17,
    22 AT 18: HLIN 17,19 AT 17: VLIN
    21,24 AT 20: VLIN 21,24 AT 1
    9: VLIN 21,22 AT 18: PLOT 21
    ,21: COLOR= 9
1222  PLOT 16,16: PLOT 20,17: PLOT
    17,20: PLOT 22,21: PLOT 21,2
    2: COLOR= 2: PLOT 18,18: PLOT
    19,19
1225 SX =  - .5:SY =  - .5
1230  RETURN
2000  IF  INT (LL *  RND (1)) + 1
    = 2 THEN 2030
2010  RETURN
2030 TY =  INT (4 *  RND (1)) + 1
    : IF TY = 1 THEN X(I) = 4:Y(
    I) =  INT (30 *  RND (1)) +
    5:SX(I) = 1:SY(I) = 0: RETURN
```

TIME PILOT

▽

```
2040  IF TY = 2 THEN X(I) = 35:Y(
      I) =  INT (30 *  RND (1)) +
      5:SX(I) =  - 1:SY(I) = 0: RETURN

2050  IF TY = 3 THEN X(I) =  INT
      (30 *  RND (1)) + 5:Y(I) = 5
      :SX(I) = 0:SY(I) = 1: RETURN

2060 X(I) =  INT (30 *  RND (1)) +
      5:Y(I) = 35:SX(I) = 0:SY(I) =
      - 1: RETURN
2100  FOR S = 1 TO 5: IF S / 2 =
      INT (S / 2) THEN  POKE  - 1
      6304,0: POKE  - 16302,0: POKE
      - 16299,0: POKE  - 16298,0:
      GOTO 2105
2102  POKE  - 16304,0: POKE  - 16
      301,0: POKE  - 16300,0: POKE
      - 16298,0
2105  POKE 768,20: POKE 769,10: CALL
      770: NEXT S
2107  FOR S = Y(I) - 2 TO Y(I) +
      2
2110  COLOR= TT: HLIN X(I) - 3,X(
      I) + 3 AT S: NEXT S:SC = SC +
      10 * V: COLOR= 15: PLOT NU,0
      :NU = NU - 1: IF NU =  - 1 THEN
      NU = 0: IF BG = 0 THEN BG =
      1:X =  INT (30 *  RND (1)) +
      5:Y = 5
2111  COLOR= TT: PLOT C(U),Z(U)
2115 C(U) = 0
2117 X(I) = 0
2120  RETURN
3000  ON V GOTO 3010,3050,3100,31
      50,3200
3010  COLOR= 4: HLIN X(I) - 1,X(I
      ) + 1 AT Y(I) - 2: VLIN Y(I)
      - 1,Y(I) + 1 AT X(I): HLIN
      X(I) - 3,X(I) + 3 AT Y(I) +
      1: HLIN X(I) - 3,X(I) + 3 AT
      Y(I) + 2: COLOR= 0
3011  PLOT X(I) + 2,Y(I) + 2: PLOT
      X(I) - 2,Y(I) + 2
3015  POKE 768, INT (5 *  RND (1)
      ) + 240: POKE 769,4: CALL 77
      0
3020  RETURN
3050  COLOR= 12: HLIN X(I) - 1,X(
      I) + 1 AT Y(I) - 2: VLIN Y(I
      ) - 1,Y(I) + 2 AT X(I): HLIN
      X(I) - 3,X(I) + 3 AT Y(I) +
      1: COLOR= 8: PLOT X(I) - 2,Y
      (I) + 1: PLOT X(I) + 2,Y(I) +

      1: COLOR= 2
3052  PLOT X(I),Y(I) + 1: FOR U =
      1 TO 3: POKE 768,241: POKE 7
      69,3: CALL 770: NEXT U
3060  RETURN
3100  COLOR= 14: HLIN X(I) - 1,X(
      I) + 1 AT Y(I) - 2: HLIN X(I
      ) - 3,X(I) + 3 AT Y(I): HLIN
      X(I) - 2,X(I) + 2 AT Y(I) +
      1: PLOT X(I),Y(I) + 2: COLOR=
      1
3105  PLOT X(I) - 3,Y(I) - 1: PLOT
      X(I) + 3,Y(I) - 1: PLOT X(I)
      ,Y(I) - 1: POKE 768, INT (30
       *  RND (1)) + 100: POKE 769
      ,10: CALL 770
3110  RETURN
3150  COLOR= 3: HLIN X(I) - 3,X(I
      ) + 3 AT Y(I) - 1: HLIN X(I)
       - 2,X(I) + 2 AT Y(I): HLIN
      X(I) - 1,X(I) + 1 AT Y(I) +
      1: PLOT X(I),Y(I) + 2: PLOT
      X(I) - 3,Y(I) - 2: PLOT X(I)
       + 3,Y(I) - 2
3155  COLOR= 9: PLOT X(I),Y(I) -
      2: COLOR= 5: VLIN Y(I),Y(I) +
      1 AT X(I): FOR U = 1 TO 3: POKE
      768,90: POKE 769,4: CALL 770
      : NEXT U
3160  RETURN
3200  COLOR= 13: HLIN X(I) - 3,X(
      I) + 3 AT Y(I) - 1: HLIN X(I
      ) - 3,X(I) + 3 AT Y(I) + 2: VLI
      Y(I),Y(I) + 1 AT X(I) - 3: VLIN
      Y(I),Y(I) + 1 AT X(I) + 3: COLOR
      2: VLIN Y(I),Y(I) + 1 AT X(I
      ) + 2: VLIN Y(I),Y(I) + 1 AT
      X(I) - 2
3205  FOR U = 1 TO 5: POKE 768,(6
      - U) * 10: POKE 769,5: CALL
      770: NEXT U
3210  COLOR= 11: HLIN X(I) - 1,X(
      I) + 1 AT Y(I): HLIN X(I) -
      1,X(I) + 1 AT Y(I) + 1: RETURN

4000  COLOR= TT: FOR I = X - 3 TO
      X + 3: VLIN Y - 3,Y + 3 AT I
      : NEXT I
4005 Y = Y + 2
4010  FOR I = 0 TO 3: COLOR=  INT
      (40 *  RND (1)) + 1: HLIN X -
      I,X + I AT Y - I: HLIN X - I
      ,X + I AT Y + I: VLIN Y - I,
      Y + I AT X + I: VLIN Y - I,Y
       + I AT X - I: NEXT I
4012  POKE 768, INT (30 *  RND (1
      )) + 10: POKE 769,7: CALL 77
      0
4015  IF Y > 38 THEN V = V + 1: POP
      : GOTO 100
4020  FOR I = 1 TO 2: IF C(I) = 0
      THEN 4050
4030  IF C(I) > X - 3 AND C(I) <
      X + 3 AND Z(I) > Y - 3 AND Z
      (I) < Y + 3 THEN 4040
4035  NEXT I: GOTO 4050
4040  FOR I = 1 TO 30: IF I / 2 =
      INT (I / 2) THEN  POKE  - 1
      6304,0: POKE  - 16302,0: POKE
      - 16299,0: POKE  - 16298,0:
      GOTO 4045
4042  POKE  - 16304,0: POKE  - 16
      301,0: POKE  - 16300,0: POKE
      - 16298,0
4045  FOR U = 1 TO 50: NEXT U,I:V
      = V + 1:SC = SC + 500: GOTO
      100
4050  IF Y > 12 AND Y < 28 AND X >
      12 AND X < 28 THEN SH = SH -
      2:KL =  PEEK ( - 16336) +  PEEK
      (49200)
4060  RETURN
5000 PL = PL - 1
5010  FOR I = 1 TO 50:KL =  PEEK
      ( - 16336) +  PEEK ( - 16336
      ) -  PEEK ( - 16336): FOR U =
      1 TO I / 2: NEXT U: COLOR= 1
      3: HLIN  INT (40 *  RND (1))
      , INT (40 *  RND (1)) AT  INT
      (40 *  RND (1))
5020  VLIN  INT (40 *  RND (1)), INT
      (40 *  RND (1)) AT  INT (40 *
       RND (1))
5025  NEXT I
5026 SH = 50
5030  IF PL = 0 THEN 5100
5035 RS = 1
5040  FOR I = 1 TO 1000: GOTO 100

5100  PRINT : PRINT : PRINT "SCOR
      E "SC"  ANOTHER ";: INPUT A$
      : IF  LEFT$ (A$,1) = "N" THEN
       END
5110  CLEAR : GOTO 5
```

WIPEOUT

▽

```
LIST

10  REM  ****************
12  REM  * GEOFF MORGAN *
14  REM  *    1983      *
16  REM  ****************
20  HOME
30  GOSUB 1500
40  VTAB 22: HTAB 8: PRINT "HELLO! I'M ---- !"
50  VTAB 24: HTAB 8: INPUT "WHAT IS YOUR NAME? ";N$
60  HOME : VTAB 22: HTAB 8: PRINT "DO YOU NEED HELP BEFORE"
70  VTAB 24: HTAB 8: PRINT "STARTING THE GAME?  (Y/N)";
80  GET Y$: IF Y$ = "Y" THEN 110
90  IF Y$ = "N" THEN 130
100  GOTO 80
110  GOSUB 1620
120  GOTO 140
130 D = 1: GOSUB 1620
140  IF D = 1 THEN 260
150  TEXT : HOME : PRINT  TAB( 5)N$ + "-";
160  PRINT : PRINT : PRINT  TAB( 5)"THE DIGIT INDICATED MUST"
170  PRINT : PRINT : PRINT  TAB( 5)"MUST BE REMOVED IN ONE MOVE."
180  PRINT : PRINT : PRINT : PRINT  TAB( 5)"FOR EXAMPLE-"
190  PRINT : PRINT  TAB( 18)"TO 'WIPE OUT' THE"
200  PRINT : PRINT  TAB( 5)"3 IN 32, 30 (3 TENS) MUST BE"
210  PRINT : PRINT  TAB( 5)"SUBTRACTED FROM 32 TO GIVE 2."
220  VTAB 24: PRINT  TAB( 6)"(PRESS 'SPACE BAR' TO CONTINUE.)";
230  GET A$: IF A$ = " " THEN 250
240  GOTO 230
250  GOSUB 1950
260  TEXT : HOME : VTAB 4: HTAB 8: PRINT N$ + "-";
270  VTAB 6: HTAB 8
280  PRINT "TO SELECT THE NUMBERS YOU"
290  PRINT : PRINT  TAB( 8)"WOULD LIKE TO WORK WITH"
300  PRINT : PRINT  TAB( 8)"TYPE THE NUMBER PRECEDING"
310  PRINT : PRINT  TAB( 8)"YOUR SELECTION."
320  PRINT : PRINT : PRINT  TAB( 12)"1. TENS"
330  PRINT  TAB( 12)"2. HUNDREDS"
340  PRINT  TAB( 12)"3. THOUSANDS"
350  PRINT  TAB( 12)"4. TEN-THOUSANDS"
360  PRINT  TAB( 12)"5. HUNDRED-THOUSANDS"
370  PRINT  TAB( 12)"6. MILLIONS"
380  GET B$
390 Z =  VAL (B$):ZZ = Z + 1
400  IF Z > 0 THEN 420
410  GOTO 430
420  IF Z < 7 THEN 450
430  PRINT : PRINT : PRINT  TAB( 7)"YOU DID NOT PRESS A NUMBER FROM 1 TO
     6. TRY AGAIN.";
440  FOR G = 1 TO 2000: NEXT : GOTO 260
450 CC = 0
460 C = 1
470  IF C = 11 THEN 1140
480 RR$ = "":M$ = "":LL$ = "":R$ = ""
490 X =  RND (2)
500 X =  INT (X * 10 ^ ZZ)
510  IF X < 10 ^ Z THEN 480
520  HOME : VTAB 12: HTAB 12
530 X$ =  STR$ (X)
540  IF Z > 2 THEN 910
550  PRINT X$
560 Y =  RND (9):Y =  INT (Y * 10)
```

```
570  IF Y = 0 THEN 560
580  IF Y = 4 THEN 560
590  IF Y = 8 THEN 560
600  IF Z < 3 THEN 650
610  IF Z = 3 THEN ZZ = 5
620  IF Z = 4 THEN ZZ = 6
630  IF Z = 5 THEN ZZ = 7
640  IF Z = 6 THEN ZZ = 9
650  IF Y > ZZ THEN 560
660 YY = Y
670  FOR T =  LEN (X$) TO 1 STEP  - 1
680 RR$ = RR$ + ( MID$ (X$,T,1)): NEXT
690 Y$ =  MID$ (RR$,Y,1)
700  IF Y$ = "0" THEN 560
710  IF Y$ = " " THEN 560
720 V =  LEN (X$) - Y
730  FOR G = 1 TO 3: VTAB 13: HTAB 12 + V: PRINT "^";
740  GOSUB 950
750  VTAB 13: HTAB 12 + V: PRINT " ": GOSUB 950
760  NEXT : VTAB 13: HTAB 12 + V: PRINT "^"
770  VTAB 16: HTAB (4): PRINT "WIPE OUT THE DIGIT MARKED BY THE '^'"
780  FOR G = 1 TO 200: NEXT
790  IF Y < 4 THEN 830
800  IF Y = 9 THEN 820
810 Y = Y - 2: GOTO 840
820 Y = Y - 3: GOTO 840
830 Y = Y - 1
840  VTAB 19: HTAB 6: PRINT "TYPE THE NUMBER"
850  HTAB 6: INPUT "TO BE SUBTRACTED - ";RR$: GOSUB 1730
860 P =  VAL (Y$):Q = P * (10 ^ Y):PP = X - Q
870  IF PP <  > INT (PP) THEN PP =  INT (PP + 1)
880 BB = X - RR
890  IF BB = PP THEN 960
900  GOTO 1290
910 R$ =  RIGHT$ (X$,3)
920  IF Z = 6 THEN 940
930 P = Z - 2:L$ =  LEFT$ (X$,P):X$ = L$ + " " + R$: GOTO 550
940 M$ =  MID$ (X$,2,3):LL$ =  LEFT$ (X$,1):X$ = LL$ + " " + M$ + " " + R
    $: GOTO 550
950  FOR G = 1 TO 500: NEXT : RETURN
960  GOSUB 2040
970  VTAB 12: HTAB 12 + V
980  IF VV <  > 0 THEN 1000
990  PRINT " ";: GOTO 1010
1000  PRINT "0";:
1010  PRINT  CHR$ (7);: FOR W = 1 TO 2000: NEXT
1020  HOME : VTAB 13: HTAB 8:P =  RND (1):P =  INT (P * 10)
1030  IF P = 1 THEN 1070
1040  IF P = 2 THEN 1080
1050  IF P = 3 THEN 1090
1060  GOTO 1020
1070  FLASH : PRINT "- - WELL DONE ";N$;" - -": GOTO 1100
1080  FLASH : PRINT "* * MARVELLOUS ";N$;" * *": GOTO 1100
1090  FLASH : PRINT "* * YOU BEAUTY ";N$;" * *": GOTO 1100
1100  FOR G = 1 TO 1000: NEXT
1110 ZZ = Z + 1:I = 0
1120  NORMAL : HOME :CC = CC + 1:RR$ = "":C = C + 1: GOTO 470
1130  GOTO 1140
1140  VTAB 12: HTAB 8: PRINT N$ + " - "
1150  HTAB 8: PRINT "GOOD WORK!"
1160  PRINT
1170  PRINT  TAB( 8)"YOU HAVE ";CC;" OUT OF ";C - 1;" CORRECT!"
```

WIPEOUT is an educational drill and practice program designed to strengthen place value skills. The student can select the magnitude of the numbers to be worked with – six levels (tens through millions) - and instructions can be called for at any point during the game.

The student is required to 'wipe out' the randomly selected digit in the randomly selected number within the range chosen. For example, to 'wipe out' the '3' in 23 576, '3000' is entered as the number to be subtracted to give 20 576.

Geoff Morgan
Ferny Hills Qld

```
1180 I = 0
1190  GOTO 1200
1200  FOR G = 1 TO 2000: NEXT G
1210  HOME : VTAB 12: HTAB 8: PRINT "PRESS 'SPACE BAR' TO CONTINUE"
1220  PRINT : PRINT : PRINT  TAB( 8)"PRESS 'E' TO END"
1230  GET G$:
1240  IF G$ = " " THEN 260
1250  IF G$ = "E" THEN 1270
1260  GOTO 1230
1270  HOME : VTAB 12: HTAB 5: PRINT "THANK YOU FOR PLAYING ";N$;
1280  END
1290  FOR W = 1 TO 3: PRINT  CHR$ (7);: NEXT
1300  VTAB 22: HTAB 6
1310 P =  RND (1):P =  INT (P * 10)
1320  IF P = 1 THEN 1360
1330  IF P = 2 THEN 1370
1340  IF P = 3 THEN 1380
1350  GOTO 1310
1360  PRINT "# # SORRY, NOT CORRECT ";N$: GOTO 1390
1370  PRINT "# # THAT'S A MISTAKE ";N$: GOTO 1390
1380  PRINT "# # YOU MISSED THAT ";N$: GOTO 1390
1390  PRINT : PRINT  TAB( 7)"TRY AGAIN! # #";
1400  FOR G = 1 TO 2000: NEXT
1410  VTAB 19: PRINT  SPC( 100);
1420  VTAB 22: HTAB 6: PRINT  SPC( 100);
1430 I = I + 1: IF I = 3 THEN 1450
1440  GOTO 840
1450  HOME : VTAB 13: HTAB 8:
1460  PRINT "THE CORRECT NUMBER TO"
1470  PRINT : PRINT  TAB( 8)"SUBTRACT IS "; VAL (Y$) * 10 ^ Y;
1480  FOR G = 1 TO 2000: NEXT
1490 I = 0:RR$ = "":C = C + 1: GOTO 470
1500  GR : COLOR= 14: FOR X = 0 TO 39: HLIN 0,39 AT X: NEXT
1510  COLOR= 1
```

WIPEOUT

▽

```
1520  VLIN 12,26 AT 2: VLIN 12,26 AT 8: HLIN 4,6 AT 12: VLIN 12,26 AT 4:
     VLIN 12,26 AT 6
1530  VLIN 23,26 AT 3: VLIN 23,26 AT 7
1540  VLIN 12,26 AT 11: VLIN 12,26 AT 13: HLIN 13,17 AT 12: HLIN 13,17 AT
    18: VLIN 12,16 AT 17
1550  VLIN 12,17 AT 17
1560  VLIN 12,26 AT 19: HLIN 19,22 AT 12: HLIN 19,20 AT 17: HLIN 19,22 AT
    26
1570  VLIN 12,26 AT 24: VLIN 12,26 AT 27: HLIN 24,27 AT 12: HLIN 24,27 AT
    26
1580  VLIN 12,26 AT 29: VLIN 12,26 AT 32: HLIN 29,32 AT 26
1590  HLIN 34,38 AT 12: VLIN 12,26 AT 36
1600  GOSUB 1670
1610  RETURN
1620  COLOR= 14: FOR X = 12 TO 26: HLIN 0,39 AT X
1630 G =  PEEK (S)
1640  FOR Y = 1 TO 100
1650  NEXT Y: NEXT X
1660  RETURN
1670 S =  - 16336
1680  FOR B = 1 TO 50
1690 G =  PEEK (S) -  PEEK (S) +  PEEK (S): NEXT
1700  FOR B = 1 TO 50
1710 G =  PEEK (S) -  PEEK (S) +  PEEK (S) -  PEEK (S) +  PEEK (S) -  PEEK
    (S) +  PEEK (S)
1720  NEXT : RETURN
1730 E =  LEN (RR$)
1740  IF RR$ = "" THEN 840
1750  FOR L = 1 TO E
1760 EE$ =  MID$ (RR$,L,1)
1770 EE =  ASC (EE$)
1780  IF EE = 32 THEN 1810
1790  IF EE < 48 THEN 1830
1800  IF EE > 57 THEN 1830
1810  NEXT L
1820 RR =  VAL (RR$): RETURN
1830  IF E <  > 1 THEN 1870
1840  IF EE = 81 THEN 1940
1850  IF EE = 82 THEN 260
1860  IF EE = 72 THEN 150
1870  FOR W = 1 TO 2: PRINT  CHR$ (7);: NEXT
1880  VTAB 22: PRINT "YOU DID NOT ENTER ";: INVERSE : PRINT "A NUMBER GRE
    ATER": NORMAL
1890  HTAB 14: INVERSE : PRINT "THAN ZERO!": NORMAL
1900  FOR W = 1 TO 2000: NEXT
1910  VTAB 22: PRINT  SPC( 80);
1920  PRINT  CHR$ (7);: VTAB 20: PRINT  SPC( 39);
1930 RR$ = "": GOTO 840
1940  HOME : GOTO 1140
1950  HOME : VTAB 8: HTAB 5: PRINT "DURING THEN GAME ENTER:"
1960  PRINT : HTAB 9: PRINT "'Q' TO QUIT"
1970  PRINT : HTAB 9: PRINT "'R' TO RETURN TO MENU"
1980  PRINT : HTAB 9: PRINT "'H' TO GET INSTRUCTIONS"
1990  VTAB 18: HTAB 4: PRINT "PRESS 'SPACE BAR' TO CONTINUE."
2000  GET G$
2010  IF G$ = " " THEN 2030
2020  GOTO 2000
2030  RETURN
2040 VV = V: RETURN

]
```

APPLE SPEED LOCK

Lots of unlocked files on your disk, and hours of typing to lock them up away from the kids? Try Speed-Lock.

The Speed-Lock will first catalog the disk, and when the end of catalog is reached, a short data POKE sequence is run, (about 6 seconds) and a menu placed at the top of the screen: Lock, Unlock, Normal, Quit. Selection of Lock will cause the drive to step through each listing on the Displayed catalog only, and lock the files. Unlock performs in the same manner. Normal simply catalogs the disk, then exits the program. Quit simply clears the screen and ends.

For disks with full catalogs, that is, more than 1 screen-full, only the last screen display will be locked. A short catalog interrupt sequence should be no problem so you can lock the first screens and then move on.

R. Chalmers
Inala Qld

```
]LIST

10  TEXT : HOME : CLEAR
20  REM                          ==========                    SPEED
     LOCK                    ==========


30  PRINT  CHR$ (4)"CATALOG"
40  DIM A(24),N$(30)
50 :
60  FOR I = 1 TO 24: READ A(I): NEXT I
70  REM IX/VII/MCMLXXXII
80 T =  PEEK (37)
90  IF T >  = 23 THEN S = 0: GOTO 110
100 S = 5
110  GOSUB 380
120  INVERSE : VTAB 1: HTAB 1: PRINT "(L)OCK  (U)NLOCK  (N)ORMAL
     (Q)UIT";: NORMAL : PRINT " ?"; CHR$ (8);: GET AN$
130  HOME
140  IF AN$ = "L" THEN 190
150  IF AN$ = "U" THEN 310
160  IF AN$ = "Q" THEN 260
170  IF AN$ = "N" THEN 270
180  GOTO 120
190  PRINT "LOCK   "
200  FOR X = S TO T
210 N$(X) =  MID$ (N$(X),7)
220  PRINT  CHR$ (4)"LOCK"N$(X)
230  VTAB 1: HTAB 5: PRINT " "N$(X)
240  NEXT X
250  GOTO 120
260  GOTO 500
270  PRINT : HOME
280  PRINT  CHR$ (4)"CATALOG"
290  GOTO 260
300  END
310  PRINT "UNLOCK   "
320  FOR X = S TO T
330 N$(X) =  MID$ (N$(X),7)
340  PRINT  CHR$ (4)"UNLOCK"N$(X)
350  VTAB 1: HTAB 7: PRINT " ";N$(X)
360  NEXT X
370  GOTO 120
380  FOR X = S TO 24
390  FOR Y = 0 TO 29
400 N$(X) = N$(X) +  CHR$ ( PEEK (A(X) + Y))
410  NEXT Y
420  IF  MID$ (N$(X),2,1) =  CHR$ (160) THEN 440
430  NEXT X
440 XX = X - 1
450  RETURN
460  DATA  1024,1152,1280,1408
470  DATA  1536,1664,1792,1920,1064,1192,1320,1448,1576,1704,1832
    ,1960,1104,1232,1360,1488,1616,1744,1872
480  DATA  2000
490  FOR X = S TO T: PRINT  LEFT$ (N$(X),1): NEXT X
500  DEL 10,490: CLEAR : END
```

HI-RES REVERSE

When using the Apple's hi-res screen, you have a whole world of graphic capabilities at your fingertips. But sometimes, as I have found, you can create a complex picture or graph and then say to yourself 'It would look a lot better if the whole screen was reversed'. Here is a short Assembly Language program that will do this for you. To utilise it simply BRUN the program after saving it to disk.

Martin Scerri
Mulgrave VIC

```
10  CLEAR : TEXT : HOME : INVERSE
     : PRINT  SPC( 40): VTAB 2: HTAB
     1: PRINT " ": VTAB 2: HTAB 4
     0: PRINT " ": VTAB 3: HTAB 1
     : PRINT  SPC( 40): NORMAL : VTAB
     2: HTAB 10: PRINT "EPSON GRA
     PHICS DUMP": VTAB 5
20  POKE 34,3
30  GOSUB 130
40  INPUT "INVERSE? (Y OR N):";IN
     $
50  INPUT "ENLARGED? (Y OR N):";E
     N$
60  IF  LEFT$ (IN$,1) = "Y" THEN
     I = 32
70  IF  LEFT$ (EN$,1) = "Y" THEN
     E = 64
80  POKE 1913,E + I + 1
90  POKE 1913,E + I + 1
100  PRINT  CHR$ (4);"PR#1"
110  PRINT  CHR$ (17)
120  PRINT  CHR$ (4);"PR#0": POKE
     34,0: END
130  PRINT  CHR$ (4);"CATALOG"
140  PRINT
150  INPUT "NAME OF HI-RES SCREEN
      TO LOAD:";RE$
160  PRINT  CHR$ (4);"BLOAD";RE$;
     ",A$2000"
170  RETURN
65535  REM ******************
65535  REM COPYRIGHT (C) 1983
65535  REM 23/7/83
65535  REM MARTIN SCERRI
65535  REM WRITTEN BY:
65535  REM ******************
```

```
*
]CALL-151

*6000L

6000-   8D 50 C0    STA   $C050
6003-   8D 52 C0    STA   $C052
6006-   8D 54 C0    STA   $C054
6009-   8D 57 C0    STA   $C057
600C-   A9 00       LDA   #$00
600E-   AB          TAY
600F-   85 F8       STA   $F8
6011-   A9 20       LDA   #$20
6013-   85 F9       STA   $F9
6015-   AA          TAX
6016-   B1 F8       LDA   ($F8),Y
6018-   49 FF       EOR   #$FF
601A-   91 F8       STA   ($F8),Y
601C-   C8          INY
601D-   D0 F7       BNE   $6016
601F-   E6 F9       INC   $F9
6021-   CA          DEX
6022-   D0 F2       BNE   $6016
6024-   60          RTS
6025-   FF          ???
```

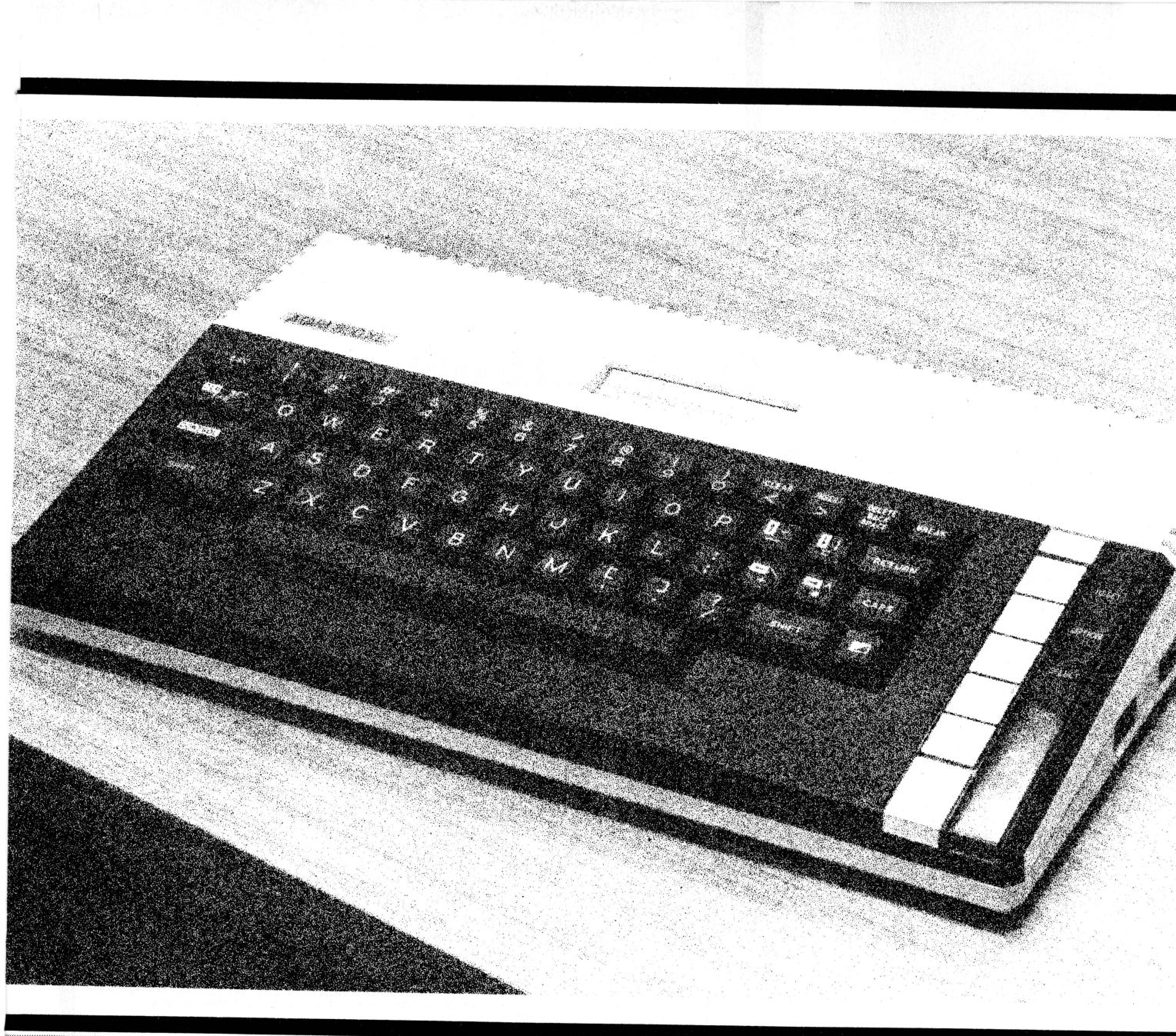
```
*
]CALL-151

*6000.6025

6000- 8D 50 C0 8D 52 C0 8D 54
6008- C0 8D 57 C0 A9 00 AB 85
6010- F8 A9 20 85 F9 AA B1 F8
6018- 49 FF 91 F8 C8 D0 F7 E6
6020- F9 CA D0 F2 60 FF
```

RESPONSE TIME

This subroutine can be included in teaching programs to gain student responses in a specified time.

Harry Klose
Wauchope NSW

```
)LIST

10  HOME : TEXT
15  VTAB 5: HTAB 1: PRINT "PLEASE
      TYPE YOUR NAME AND PRESS TH
    E     RETURN KEY"
20  PRINT  CHR$ (7): REM  - THIS
     SIGNIFIES THE BEGINNING OF 3
     SECONDS FOR RESPONSE
25 N = N + 1
30 X =  PEEK ( - 16384)
40  POKE  - 16384,0
50  IF N = 120 GOTO 300
51  REM  VALUE OF N CAN BE VARIED
      ACCORDING TO TIME REQUIRED,
      THE VALUE HERE IS ABOUT 3 S
    ECONDS
60  IF X < 128 GOTO 25
65  IF X > 127 GOTO 200
200  VTAB 8: HTAB 1: INPUT "";A$
210  VTAB 20: HTAB 1: PRINT "THAN
    K YOU ";A$
250  STOP
300  FOR I = 1 TO 3: PRINT  CHR$
    (7): NEXT : PRINT "YOU WERE
    TOO SLOW"
301  REM  - CHR$(7) SIGNIFIES THA
    T TIME IS UP
999  PRINT
1000  REM :THIS SUBROUTINE CAN BE
      INCLUDED IN TEACHING PROGRA
    MS TO GAIN STUDENT RESPONSES
      IN A SPECIFIED TIME.
1009  PRINT
1010  REM :THIS PROGRAM CREATED B
    Y HARRY KLOSE 1 MAY 1982
```



PROGRAMS FOR ATARI

COPY SELECTOR

Have you ever wished that you didn't have to enter DOS to search disk directories for a particular program? How about single key input to run the program once found?

Selector was written with both of these ideas in mind to allow my children easier access to our games disks. It will accept up to 50 filenames from a disk, although this may be increased to the ATARI's maximum of 64 by changing the DIM FILE value in line 600. 50 should be sufficient for most disks.

SAVEd programs will be RUN upon selection and LISTed programs will be ENTERed. I have assumed that LST will be used as a filename extender on LISTed programs, as recommended in the DOS Reference Manual. A load will be attempted for any file selected, however, an error message results if the load is unsuccessful (for example, if trying to load DOS.SYS).

I have made this program an AUTORUN.SYS file on all of my disks so that my children can easily select their program as soon as the disk boots, or simply read directories until they find the program they wish to run. This was achieved using the program from Automate Your ATARI, written by JJ Wrobel and published in the January 1983 issue of COMPUTE magazine, page 146.

J. Trigge
South Penrith NSW

```
 1 REM
 2 REM  SELECTOR
 3 REM  by JOHN TRIGGE
 4 REM  3 September 1983
 5 REM
10 DIM FILE$(550),IN$(17),OUT$(14),T$(1),MSG$(80),TEMP$(11)
15 GOSUB 600:REM Initialisation
20 M=1:N=64:GOSUB 500:REM Get directory
30 IF C=1 THEN POSITION 14,10:? " Disk empty ":GOTO 70
40 ? "<";CHR$(M+N);"> ";FILE$(M*11-10,M*11)
50 IF M+N=77 THEN POKE 82,22:POSITION 22,2:REM Start second column
60 M=M+1:IF M<C AND M/27<>INT(M/27) THEN 40
70 POKE 82,2:GOSUB 400:REM Get message
80 GOSUB 300:REM Sound, move message and get input
90 IF (A<49 OR A>50) AND (A<65 OR A>M+63) THEN 80
100 IF A=49 AND M<C THEN GOSUB 600:N=N-26:GOTO 40:REM More programs
110 IF A=49 THEN GOSUB 430:GOSUB 300:GOTO 70:REM No more programs
120 IF A=50 THEN GOSUB 410:GOSUB 300:GOSUB 600:GOTO 20:REM New disk
130 D=A-N:OUT$="D:"
140 TEMP$=FILE$(D*11-10,D*11-3)
150 FOR I=1 TO 8:IF TEMP$(I,I)<>" " THEN OUT$(LEN(OUT$)+1)=TEMP$(I,I):NEXT I
160 TEMP$=FILE$(D*11-2,D*11)
170 IF TEMP$(1,1)<>" " THEN OUT$(LEN(OUT$)+1)=".":REM Filename extender used
180 FOR I=1 TO 3:IF TEMP$(I,I)<>" " THEN OUT$(LEN(OUT$)+1)=TEMP$(I,I):NEXT I
190 IF OUT$(LEN(OUT$)-2)="LST" THEN GRAPHICS 0:POKE 82,2:GOSUB 450:ENTER OUT$
200 TRAP 210:GOSUB 450:RUN OUT$
210 GOSUB 420:GOSUB 300
220 POSITION 7,19:? "(30 spaces)"
230 GOTO 70
300 FOR Z=1 TO 2:FOR Y=1 TO 40
310 SOUND 0,96,10,10:NEXT Y
320 SOUND 0,0,0,0:NEXT Z
330 POSITION 1,23:? MSG$(1,38);
340 T$=MSG$(1,1):MSG$=MSG$(2):MSG$(LEN(MSG$)+1)=T$:REM Move message
350 FOR W=1 TO 40:NEXT W
360 IF PEEK(764)=255 THEN 330
370 OPEN #2,4,0,"K:":REM Open keyboard
380 GET #2,A:POKE 764,255
390 CLOSE #2:RETURN
400 MSG$="*PRESS LETTER BESIDE DESIRED PROGRAM**1=MORE PROGRAMS*2=NEW
    DISK*":RETURN
410 MSG$="*PLACE NEW DISK IN DRIVE**PRESS ANY KEY TO CONTINUE*":RETURN
420 MSG$="*PROGRAM NOT AVAILABLE FROM BASIC**PRESS ANY KEY TO CONTINUE*":RETURN
430 MSG$="*NO MORE FILES ON THIS DISK**PRESS ANY KEY TO CONTINUE*":RETURN
450 POSITION 7,19
460 ? "Searching for ";OUT$
470 RETURN
500 OPEN #1,6,0,"D:*.*":REM Directory
510 C=1:FILE$=" "
520 INPUT #1,IN$:REM Input filename
530 IF IN$(LEN(IN$)-11,LEN(IN$))="FREE SECTORS" THEN 570
540 IN$=IN$(3,13):REM Remove 'locked'
550 FILE$(C*11-10,C*11)=IN$
560 C=C+1:GOTO 520
570 CLOSE #1:RETURN
600 POKE 82,2:GRAPHICS 0
610 SETCOLOR 2,1,1:POKE 752,1
620 ? "(11 spaces)DISK DIRECTORY(11 spaces)"
630 POKE 82,3:?
640 RETURN
```

PROGRAMS FOR COMMODORE 64

WORD TEASER

Word Teaser is a game designed to trick your friends into believing that you have one of the most intelligent computer in the world.

When you have keyed in the program and run it you will be confronted by the question "Are you ready to start?" If you type in yes the game will start, if you type in no the program will end. If you type in an asterisk instructions will be displayed on the screen.

After you have read the instructions try the program on your friends, they will be amazed.

Jarrad Webb
Henley Beach SA

```
1 DIMS$(50),X(50):X=1
5 PRINT"■"
50 PRINT" *   *  ***  ****  ****"
52 PRINT" *   * *   * *   * *   *   *********"
54 PRINT" * * * *   * ****  *   *  *BY J.WEBB*"
56 PRINT" * * * *   * *  *  *   *   *********"
58 PRINT"  * *   ***  *   * ****"
60 PRINT
62 PRINT" *****  ****  ***   ****  *** ****"
64 PRINT"   *   *     *   * *     *    *   *"
66 PRINT"   *   ***   *   *  ***  ***  ****"
68 PRINT"   *   *     *****     * *    *  *"
70 PRINT"   *    **** *   * ****   *** *   *"
80 PRINT"■■PLEASE GUESS A WORD (IT MUST BE A NOUN)"
90 PRINT"■■ARE YOU READY TO START■ (Y/N) ■"
100 GETA$:IFA$=""THEN100
105 IFA$="*"THEN300
110 IFA$="Y"THEN150
120 IFA$="N"THENPRINT"■■SEE YOU LATER,BYE":END
130 GOTO100
150 PRINT"■■■■■■■■■■■■■■■■WORD TEASER■"
160 INPUT"■WHAT COLOUR IS IT ";0$:GOSUB500
```

GRAPH

This program uses the HIRES machine code routines to plot graphs of user-supplied functions within a range of values specified by the user. The desired function is simply input; the program contains a mini-interpreter to code the function and allow the X and Y coordinates to be calculated for the plot.

The plot is automatically scaled, graduated and labelled, X and Y axes are displayed, and the title and range indications are displayed. The program allows functions to be plotted which include any of the normal arithmetic operators as well as SIN, COS, TAN, ATN, EXP, LOG, SQR, ABS and INT. The exponentiation 'up arrow' and parentheses may also be used.

The range of X values required should not include any values for which the function is undefined (eg. division by zero).

The HIRES routines must be loaded into memory before this program can be run. Take great care when typing this program into your computer that there are exactly 59 spaces between the '=' and the '2' in the first line.

M. Griffiths
Lindisfarne NSW

```
100 DEFFNA(X)=                                                    2*X+3
101 REM
102 REM   MIKE GRIFFITHS - 1983
103 REM
104 J=49581:POKE49681,14
105 FLAG=FLAG+1:IFFLAG>1THEN295
106 PRINTCHR$(144)
110 DIMY(300):DIMX(300):DIMD$(14):DIMN$(14):GOSUB5000:GOTO180
120 XH=INT(X1/256):XL=X1-256*XH:POKEJ,Y1:POKEJ+1,XL:POKEJ+2,XH
140 SYS49711
150 RETURN
180 POKE53281,14:PRINTCHR$(147):PRINT"SPECIFY FUNCTION IN THE FORM Y=F(X)"
185 PRINT:PRINT"YOU CAN INCLUDE":PRINT
187 FORX=1TO3:PRINTD$(X)TAB(10)D$(X+3)TAB(20)D$(X+6):NEXT
188 PRINT:PRINT"(ENTER ZERO TO END)"
190 PRINT:INPUT"Y=";F$
192 IFF$="0"THENEND
195 PRINTCHR$(147):PRINT:PRINTTAB(12)"CODING FUNCTION"
200 SD$="Y="+F$
210 A=LEN(F$)-2
220 FORX=1TOA
230 T$=MID$(F$,X,3)
240 FORY=1TO9
250 IFT$=D$(Y)THENF$=LEFT$(F$,X-1)+N$(Y)+RIGHT$(F$,A-X):A=A-2
260 NEXT:NEXT
262 A=LEN(F$)
264 FORX=1TOA
266 T$=MID$(F$,X,1)
268 FORY=10TO14
270 IFT$=D$(Y)THENF$=LEFT$(F$,X-1)+N$(Y)+RIGHT$(F$,A-X)
272 NEXT:NEXT
274 IFLEN(F$)<64THENF$=" "+F$:GOTO274
275 IFLEN(F$)>64THENPRINTCHR$(147):PRINT"FUNCTION TOO LONG":END
276 FORX=1TOLEN(F$)
280 POKE2059+X,ASC(MID$(F$,X,1))
290 NEXT
292 GOTO100
295 PRINTCHR$(147):PRINT:PRINT"INDICATE RANGE OF PLOT"
300 PRINT:INPUT"LOWEST X VALUE ";L
310 PRINT:INPUT"HIGHEST X VALUE";H
315 PRINTCHR$(147):PRINT:PRINTTAB(11)"CALCULATING POINTS"
```

```
165 INPUT"■IS IT MADE OF WOOD ";Q$:GOSUB500
166 INPUT"■IS IT MADE OF PLASTIC ";Q$:GOSUB500
167 INPUT"■IS IT MADE OF METAL ";Q$:GOSUB500
168 INPUT"■IS IT MADE OF NATURAL SUBSTANCES ";Q$:GOSUB500
170 INPUT"■DOES IT EAT FOOD ";Q$:GOSUB500
175 INPUT"■IS IT VERY LARGE ";Q$:GOSUB500
180 INPUT"■IS IT VERY HEAVY ";Q$:GOSUB500
185 INPUT"■DOES IT BREAK ";Q$:GOSUB500
190 INPUT"■DOES IT MAKE A NOISE ";Q$:GOSUB500
195 INPUT"■IS IT ANIMAL,VEGETABLE OR MINERAL ";Q$:GOSUB500
200 INPUT"■IS IT NATURAL ";Q$:GOSUB500
205 INPUT"■DOES IT GIVE OFF AN ODOUR ";Q$:GOSUB500
210 INPUT"■DOES IT MOVE ";Q$:GOSUB500
220 PRINT"■I'M SORRY I DON'T KNOW WHAT IT IS"
    :PRINT"■PLEASE TRY AGAIN"
230 FORT=1TO5000:NEXT:RUN
300 PRINT"■■■DO YOU WANT TO SEE THIS <Y/N> ?"
305 GETA$:IFA$=""THEN305
310 IFA$="N"THENRUN
315 IFA$<>"Y"THEN305
320 PRINT"■■■-ASK YOUR FRIENDS TO TELL YOU A NOUN."
321 PRINT"■-THEN ANSWER THE QUESTIONS ACCORDINGLY."
322 PRINT"■-AFTER YOU HAVE ANSWERED EACH QUESTION"
323 PRINT"■-PRESS <RETURN>."
324 PRINT"■-THEN WHEN NO-ONE IS LOOKING ENTER"
325 PRINT"■-LETTERS OF WORD ONE AFTER EACH RETURN."
326 PRINT"■-WHEN ALL THE LETTERS HAVE BEEN"
327 PRINT"■-ENTERED PRESS THE <1> KEY."
328 PRINT"■-THE WORD WILL THEN APPEAR ON THE "
329 PRINT"■-SCREEN AS IF BY MAGIC."
330 PRINT"■■■■■■■■■■■■ANY KEY TO RE-RUN■"
332 GETA$:IFA$=""THEN332
335 GOTO5
500 GETA$:IFA$=""THEN500
505 IFA$="↑"THENGOTO600
510 S$(X)=A$
520 X=X+1
530 RETURN
600 PRINT"■■■I THINK I'VE GOT IT.IT'S A ....."PRINT
610 FORT=1TOX:PRINTS$(T);
620 FORY=1TO500:NEXTY
630 NEXTT
640 PRINT"■■■■■■■■■■■ANOTHER GO <Y/N>"
650 GETA$:IFA$=""THEN650
660 GOTO110
```

```
320 N=0
330 IFL>HTHENT=H:H=L:L=T
335 I=(H-L)/280*.99999999
340 FORX=LTOHSTEPI
360 N=N+1:X(N)=N+19:Y(N)=FNA(X)
370 NEXT
380 S=Y(1):B=Y(1)
385 PRINTCHR$(147):PRINT:PRINTTAB(16)"SCALING"
390 FORX=1TON
400 IFY(X)<STHENS=Y(X)
410 IFY(X)>BTHENB=Y(X)
420 NEXT
425 IF(B-S)<.00001THENS=S-5:B=B+5
430 R=160/(B-S)
435 SYS49584:POKE253,1
440 X1=19:Y1=20:X2=19:Y2=180:GOSUB7000
450 X1=18:X2=18:GOSUB7000
460 X1=18:X2=300:Y1=181:Y2=181:GOSUB7000
470 Y1=182:Y2=182:GOSUB7000
480 FORY1=20TO180STEP16
485 X1=16:GOSUB120:X1=17:GOSUB120
490 NEXT
500 FORX1=20TO300STEP28
505 Y1=183:GOSUB120:Y1=184:GOSUB120
510 NEXT
560 SX=5:SY=0:GOSUB1000
570 SD$="X RANGE "+STR$(L)+" TO "+STR$(H)
580 SX=5:SY=2:GOSUB1000
590 SD$="Y RANGE "
600 S$=STR$(S)
610 GOSUB3000
620 SD$=SD$+S$+" TO "
630 S$=STR$(B)
640 GOSUB3000
650 SD$=SD$+S$
660 SX=5:SY=3:GOSUB1000
670 SD$="Y"
680 SX=1:SY=11:GOSUB1000
690 SD$="X"
692 SX=21:SY=23:GOSUB1000
694 FORX=1TON
696 Y(X)=INT((180-(Y(X)-S)*R)+.5)
698 X2=X1:Y2=Y1:X1=X(X):Y1=Y(X)
699 IFX>1THENGOSUB7000
700 NEXT
703 SD$="PRESS SPACE          TO EXIT"
706 SX=6:SY=24:GOSUB1000
710 GETM$:IFM$=""THEN710
720 POKE53272,21:POKE53265,27:PRINT"■":GOTO180
1000 POKE56334,PEEK(56334)AND254
1003 POKE1,PEEK(1)AND251
1007 FORA=1TOLEN(SD$)
1010 PC=ASC(MID$(SD$,A,1))
1015 IFPC=32THEN1055
1020 IFPC>61THENPC=PC-64
1030 FORBY=0TO7
1040 POKE8192+SX*8+320*SY+BY,PEEK(53248+8*PC+BY)
1050 NEXT
1055 SX=SX+1
1060 NEXT
1063 POKE1,PEEK(1)OR4
1067 POKE56334,PEEK(56334)OR1
1070 RETURN
3000 A=1
3010 DP$=MID$(S$,A,1)
3020 IFDP$="."THENNC=A-1:GOTO3040
3030 A=A+1:IFA<LEN(S$)THEN3010
3035 D=0:GOTO3050
3040 D=3-NC:IFD<0THEND=0
3050 Q=VAL(S$)
3060 D=10↑D
3070 Q=INT((Q+.5/D)*D)/D
3080 S$=STR$(Q):RETURN
5000 FORX=1TO14
5010 READD$(X),N:N$(X)=CHR$(N)
5020 NEXT
5030 RETURN
6000 DATA "INT",181,"ABS",182,"SQR",186,"LOG",188,"EXP",189,"COS",190
6010 DATA "SIN",191,"TAN",192,"ATN",193,"+",170,"-",171,"*",172,"/",173
6020 DATA "↑",174
7000 XH=INT(X1/256):XL=X1-256*XH
7010 POKEJ,Y1:POKEJ+1,XL:POKEJ+2,XH
7020 XH=INT(X2/256):XL=X2-256*XH
7030 POKEJ-3,Y2:POKEJ-2,XL:POKEJ-1,XH
7040 SYS49791:RETURN
```

```
20 REM "R"=RVS ON
21 REM
22 REM "■"=RVS OFF
23 REM
24 REM "D"=DOWN
25 REM
26 REM "J"=RIGHT
27 REM
30 D=9:AD=9
35 RE=54272
40 W=17:DI=-1:VO=15
45 DIMH(55):DIML(55):DIMD(15)
50 POKE53280,10:POKE53281,11
55 GOSUB1000
60 GOSUB8000
70 POKERE+24,VO
80 POKE1145,81:POKE55417,5
100 GETM$:IFEC>0THENGOSUB500
105 IFM$=""THEN100
120 M=ASC(M$):IFM>132ANDM<137THENGOSUB1500:GOTO100
150 M=M-42:IFM<00RM>52THEN100
170 POKERE+5,AD:POKERE+6,SR:POKERE+1,H(M):POKERE,L(M):POKERE+4,W
180 CL=D(A)+D(D):FORX=1TOCL:NEXT:POKERE+4,W-1:GOTO100
500 VO=VO+DI:IFVO>15THENVO=15:DI=-1
520 IFVO<ECTHENVO=EC:DI=1
530 POKERE+24,VO
540 RETURN
1000 FORX=0TO52
1020 READH(X),L(X)
1030 NEXT
1040 FORX=0TO15
1050 READD(X)
1060 NEXT
1070 RETURN
1500 IFM=133THENFORX=0TO24:POKERE+X,0:NEXT:PRINTCHR$(147) END
1510 IFM=134THENPOKE1145+80*N,32:N=N-1:GOSUB2000:RETURN
1520 IFM=135THENGOSUB2500:RETURN
1530 IFM=136THENPOKE1145+80*N,32:N=N+1:GOSUB2000:RETURN
2000 IFN<0THENN=9
2010 IFN>9THENN=0
2020 POKE1145+N*80,81:POKE55417+N*80,5
2030 RETURN
2500 POKE55417+N*80,14
2510 PRINTCHR$(19)CHR$(154)TAB(31)"RF1       "
2515 PRINTTAB(31)"RF3 INCR."
2520 PRINTTAB(31)"RF5 OK    "
2525 PRINTTAB(31)"RF7 DECR."
2530 ON(N+1)GOSUB3000,3500,4000,4500,5000,5500,6000,6500,7000,7500
2535 IFN=40RN=50RN=8THENLP=0:HP=0
2540 PRINTCHR$(19)CHR$(30)TAB(31)"RF1 END   "
2545 PRINTTAB(31)"RF3 UP    "
2550 PRINTTAB(31)"RF5 ALTER"
2555 PRINTTAB(31)"RF7 DOWN  "
2560 POKE55417+N*80,5
2570 RETURN
3000 GOSUB3300
3010 ONMGOTO3020,3040,3050
3020 A=A+1:IFA>15THENA=15
3030 GOTO3060
3040 RETURN
3050 A=A-1:IFA<0THENA=0
3060 POKE1158,32:POKE1159,32:FORX=0TO2:POKE1209+X,32:NEXT
3070 PRINTCHR$(19)CHR$(144)TAB(13)"DDD";A
3080 AD=A*16+D
3090 PRINTCHR$(19)TAB(24)"DDDD";AD
3100 GOTO3000
3300 GETM$:IFM$=""THEN3300
3310 M=ASC(M$)
3320 IFM<134ORM>136THEN3300
3330 M=M-133:RETURN
3500 GOSUB3300
```

SYNTHONY

Synthony is a music/sound effects experimentation program which converts the Commodore 64's keyboard into a three-octave musical keyboard with full notes on the 'QWERTY' and bottom rows and half notes where appropriate on the intermediate rows.

The program uses voice 1 of the Commodore 64 and allows the user to alter the ADSR envelope, waveform and other parameters by simple manipulation of the special function keys. Formatted screen output makes these changes and displays the appropriate sound chip registers and the values currently being POKEd into them. The program can thus be used for simple entertainment, including playing tunes and experimenting with sound effects, or for developing tunes or sound effects for subsequent use in games, programs, etc.

Independent selection and alteration of attack, decay, sustain and release is provided, as well as waveform and echo selection. When pulse waveform is selected, provision is made for independent alteration of high and low pulse width. Echo can be selected and is implemented by simple modulation of the volume register. Note duration is automatically adjusted according to the attack and decay settings. The purpose of function keys and current status of all of the above parameters are displayed on the screen.

The musical note equivalent of keys and the corresponding high and low frequency values are shown in the first table. Refer to your Commodore 64 Users Manual for the values required for notes outside this octave range. The second table suggests some combinations to try.

M. Griffiths
Lindisfarne NSW

ATTACK	DECAY	SUSTAIN	RELEASE	WAVEFORM	ECHO
1	1	9	9	PULSE HI 15 LO 125	Ø
1	1	13	13	PULSE HI Ø LO 255	1Ø
1Ø	Ø	8	8	SAW TOOTH	Ø
Ø	8	Ø	Ø	PULSE HI 2 LO 2ØØ	Ø
Ø	8	Ø	Ø	NOISE	Ø
13	8	12	12	NOISE	1Ø
9	Ø	Ø	Ø	PULSE HI 1 LO Ø	Ø
Ø	Ø	15	15	TRIANGLE	5
9	9	9	9	PULSE HI 15 LO Ø	Ø

High	Low	Key	Note	High	Low	Key	Note
4	73	Z	C-2	13	156	3	G#-3
4	139	S	C#-2	14	1Ø7	E	A-3
4	2Ø8	X	D-2	15	7Ø	4	A#-3
5	25	D	D#-2	16	47	R	B-3
5	1Ø3	C	E-2	17	37	T	C-4
5	185	V	F-2	18	42	6	C#-4
6	16	G	F#-2	19	63	Y	D-4
6	1Ø8	B	G-2	2Ø	1ØØ	7	D#-4
6	2Ø6	H	G#-2	21	154	U	E-4
7	53	N	A-2	22	227	I	F-4
7	163	J	A#-2	24	63	9	F#-4
8	23	M	B-2	25	177	O	G-4
8	147	,	C-3	27	56	Ø	G#-4
9	21	L	C#-3	28	214	P	A-4
9	159	.	D-3	3Ø	141	+	A#-4
1Ø	6Ø	:	D#-3	32	94	@	B -4
1Ø	2Ø5	/	E-3	34	75	*	C-5
11	114	Q	F-3	36	85	£	C#-5
12	32	2	F#-3	38	126	↑	D-5
12	216	W	G-3				

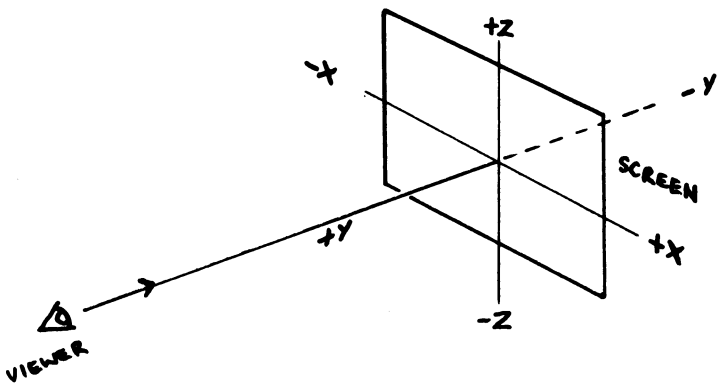
SYNTHONY

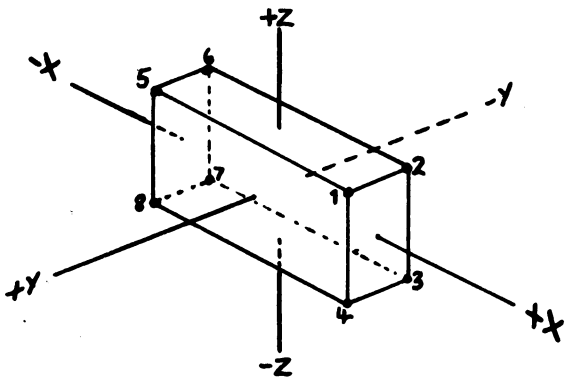
▽

```
3510 ONMGOTO3520,3540,3550
3520 D=D+1:IFD>15THEND=15
3530 GOTO3560
3540 RETURN
3550 D=D-1:IFD<0THEND=0
3560 POKE1238,32:POKE1239,32
3570 FORX=0TO2:POKE1209+X,32:NEXT
3580 PRINTCHR$(19)CHR$(144)TAB(13)"■■■■■";D
3590 AD=A*16+D
3600 PRINTCHR$(19)TAB(24)"■■■■";AD
3610 GOTO3500
4000 GOSUB3300
4010 ONMGOTO4020,4040,4050
4020 S=S+1:IFS>15THENS=15
4030 GOTO4060
4040 RETURN
4050 S=S-1:IFS<0THENS=0
4060 POKE1318,32:POKE1319,32
4070 FORX=0TO2:POKE1369+X,32:NEXT
4080 PRINTCHR$(19)CHR$(144)TAB(13)"■■■■■■■■";S
4090 SR=S*16+R
4100 PRINTCHR$(19)TAB(24)"■■■■■■■■■";SR
4110 GOTO4000
4500 GOSUB3300
4510 ONMGOTO4520,4540,4550
4520 R=R+1:IFR>15THENR=15
4530 GOTO4560
4540 RETURN
4550 R=R-1:IFR<0THENR=0
4560 POKE1398,32:POKE1399,32
4570 FORX=0TO2:POKE1369+X,32:NEXT
4580 PRINTCHR$(19)CHR$(144)TAB(13)"■■■■■■■■■■";R
4590 SR=S*16+R
4600 PRINTCHR$(19)TAB(24)"■■■■■■■■■";SR
4610 GOTO4500
5000 FORX=0TO2:POKE1649+X,32:NEXT
5010 PRINTCHR$(19)CHR$(30)TAB(13)"■■■■■■■■■■■■ ON"
5020 PRINTCHR$(154)TAB(13)"■■OFF"
5030 PRINTCHR$(144)TAB(13)"■■  "TAB(25)"17"
5040 PRINTTAB(13)"■■  "
5050 PRINTTAB(13)CHR$(154)"■■OFF"
5060 W=17:POKE54274,0:POKE54275,0
5070 RETURN
5500 FORX=0TO2:POKE1649+X,32:NEXT
5510 PRINTCHR$(19)CHR$(154)TAB(13)"■■■■■■■■■■■■■OFF"
5520 PRINTCHR$(30)TAB(13)"■■ ON"
5530 PRINTCHR$(144)TAB(13)"■■  "TAB(25)"33"
5540 PRINTTAB(13)"■■  "
5550 PRINTCHR$(154)TAB(13)"■■OFF"
5560 W=33:POKE54274,0:POKE54275,0
5570 RETURN
6000 GOSUB6300
6010 GOSUB3300
6020 ONMGOTO6030,6050,6060
6030 HP=HP+1:IFHP>15THENHP=15
6040 GOTO6070
6050 POKE54275,HP:RETURN
6060 HP=HP-1:IFHP<0THENHP=0
6070 POKE1637,32:POKE1638,32
6080 PRINTCHR$(19)CHR$(144)TAB(12)"■■■■■■■■■■■■■■■■";HP
6090 GOTO6010
6300 W=65
6310 FORX=0TO2:POKE1649+X,32:NEXT
```

```
6315 PRINTCHR$(19)CHR$(144)TAB(25)"QQQQQQQQQQQQQQQQ55 "
6320 PRINTCHR$(19)CHR$(154)TAB(13)"QQQQQQQQQQQQRQFF"
6330 PRINTTAB(13)"QRQFF"
6340 PRINTTAB(13)"QQQQQRQFF"
6350 RETURN
6500 GOSUB6300
6510 GOSUB3300
6520 ONMGOTO6530,6550,6560
6530 LP=LP+10:IFLP>255THENLP=255
6540 GOTO6570
6550 POKE54274,LP:RETURN
6560 LP=LP-10:IFLP<0THENLP=0
6570 POKE1717,32:POKE1718,32:POKE1719,32
6580 PRINTCHR$(19)CHR$(144)TAB(12)"QQQQQQQQQQQQQQQQQQ";LP
6590 GOTO6510
7000 FORX=0TO2:POKE1649+X,32:NEXT
7010 PRINTCHR$(19)CHR$(154)TAB(13)"QQQQQQQQQQQRQFF"
7020 PRINTTAB(13)"QRQFF"
7030 PRINTCHR$(144)TAB(13)"Q0  "TAB(25)"129"
7040 PRINTTAB(13)"Q0  "
7050 PRINTCHR$(30)TAB(13)"QR ON"
7060 W=129:POKE54274,0:POKE54275,0
7070 RETURN
7500 GOSUB3300
7510 ONMGOTO7520,7540,7560
7520 EC=EC+1:IFEC>10THENEC=10
7530 GOTO7570
7540 IFEC=0THENVO=15:POKERE+24,VO
7550 RETURN
7560 EC=EC-1:IFEC<0THENEC=0
7570 POKE1878,32:POKE1879,32
7580 PRINTCHR$(19)CHR$(144)TAB(13)"QQQQQQQQQQQQQQQQQQQQQQQQ";EC
7590 GOTO7500
8000 PRINTCHR$(147)CHR$(5)"QQQ11JRATTACK  ■"CHR$(144)"   0"
8010 PRINTTAB(19)CHR$(5)"R54277■"CHR$(144)" 9"
8020 PRINTTAB(3)CHR$(5)"RDECAY   ■"CHR$(144)"   9"
8030 PRINT:PRINTTAB(3)CHR$(5)"RSUSTAIN ■"CHR$(144)"   0"
8040 PRINTTAB(19)CHR$(5)"R54278■"CHR$(144)" 0"
8050 PRINTTAB(3)CHR$(5)"RRELEASE ■"CHR$(144)"  0"
8060 PRINT:PRINTTAB(3)CHR$(158)"RTRIANGLE■"CHR$(30)"  R ON"
8070 PRINT:PRINTTAB(3)CHR$(158)"RSAWTOOTH■"CHR$(154)"  RQFF"
8080 PRINT:PRINTTAB(3)CHR$(158)"RPULSE HI■  "CHR$(144)"0     ";
8085 PRINTCHR$(158)"R54276"CHR$(144)"■ 17"
8090 PRINT:PRINTTAB(3)CHR$(158)"RPULSE LO■  "CHR$(144)"0"
8100 PRINT:PRINTTAB(3)CHR$(158)"RNOISE   ■"CHR$(154)"  RQFF"
8110 PRINT:PRINTTAB(3)CHR$(5)"RECHO    ■   "CHR$(144)"0"
8120 PRINTCHR$(19)CHR$(30)TAB(31)"RQF1 END  "
8130 PRINTTAB(31)"RF3 UP   "
8140 PRINTTAB(31)"RF5 ALTER"
8150 PRINTTAB(31)"RF7 DOWN "
8160 RETURN
9000 DATA 34,75,30,141,8,147,0,0,9,159
9002 DATA 10,205,27,56,0,0,12,32,13,156
9004 DATA 15,70,0,0,18,42,20,100,0,0
9006 DATA 24,63,10,60,0,0,0,0,0,0,0,0
9008 DATA 0,0,32,94,0,0,6,108,5,103
9010 DATA 5,25,14,107,0,0,6,16,6,206
9012 DATA 22,227,7,163,0,0,9,21,8,23
9014 DATA 7,53,25,177,28,214,11,114
9016 DATA 16,47,4,139,17,37,21,154
9018 DATA 5,185,12,216,4,208,19,63,4,73
9020 DATA 0,0,36,85,0,0,38,126
9030 DATA 2,8,16,24,38,56,68,80,100,250,500,800,1000,3000,5000,8000
```

■





exceed +50 or –50 in magnitude, otherwise subsequent rotation of the shape can produce somewhat strange results (the program will not crash however, even under these circumstances). If the coordinates are initially too large or too small, the object can be scaled (eg. a scaling factor of 2 will double the magnitude of all coordinates, whereas 0.5 will halve them). The object rotates about the point X=0, Y=0, Z=0 so that positive and negative coordinates are usually required.

If the example block shape above is to be a total of 40 units long in the X direction, 12 units in the Y direction and 20 units in the Z direction, then the coordinates of its eight vertices will be:

Vertex	X	Y	Z
1	20	6	10
2	20	-6	10
3	20	-6	-10
4	20	6	-10
5	-20	6	10
6	-20	-6	10
7	-20	-6	-10
8	-20	6	-10

2) The program must then know which of the above vertices are to be joined by straight lines when the object is displayed. This is accomplished by supplying a series of 'edge indices'. In the block example, the indices could be:

-1,2,3,4,1,5,6,7,8,5,-2,6,-3,7,-4,8 – a total of 16 indices.

The magnitude of the index corresponds to one of the vertex numbers above, and a negative index indicates 'Do not draw to this point'. Taken in stages, the indices above will produce the following action:

```
 1   do not draw
 2   draw from 2 to 1
 3   draw from 3 to 2
 4   draw from 4 to 3
 1   draw from 1 to 4
 5   draw from 5 to 1
 6   draw from 6 to 5
 7   draw from 7 to 6
 8   draw from 8 to 7
 5   draw from 5 to 8
-2   do not draw
 6   draw from 6 to 2
-3   do not draw
 7   draw from 7 to 3
-4   do not draw
 8   draw from 8 to 4
```

In each case, if the index is positive, a line is drawn from that vertex to the previous one. The first index must always be negative because there is no 'previous one' in this case.

The program allows for up to 96 indices to be supplied, but the speed of drawing is improved by minimizing the number of indices (ie. minimize the number of 'do not draw').

M. Griffiths
Lindisfarne NSW

```
410 ONMGOTO330,3000,420,150,700
420 PRINTCHR$(147):PRINT:INPUT"ENTER SCALING FACTOR";F
430 IFF<1THEN500
435 F2=0
440 FORI=1TONV
450 IFABS(X(I)*F)>50THENF2=1
460 IFABS(Y(I)*F)>50THENF2=1
470 IFABS(Z(I)*F)>50THENF2=1
480 NEXT
490 IFF2>0THENPRINT"SCALING FACTOR TOO HIGH":PRINT:GOTO350
500 FORI=1TONV
510 X(I)=X(I)*F
520 Y(I)=Y(I)*F
530 Z(I)=Z(I)*F
540 NEXT
550 GOTO200
700 END
1000 PRINTCHR$(147):PRINT:PRINT"INSERT DATA TAPE THEN PRESS RETURN"
1010 GETM$:IFM$<>CHR$(13)THEN1010
1020 OPEN1,1,0,"SHAPE"
1030 INPUT#1,NV
1040 FORI=1TONV
1050 INPUT#1,X(I)
1060 INPUT#1,Y(I)
1070 INPUT#1,Z(I)
1080 NEXT
1090 INPUT#1,NE
1100 FORI=1TONE
1110 INPUT#1,E(I)
1120 NEXT
1130 CLOSE1
1140 RETURN
```

your computer

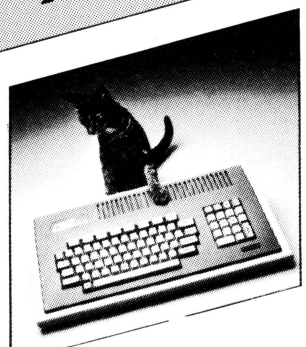
MAGAZINE **FOR BUSINESS AND PLEASURE**

$2.50

Incorporating "Your Business Computer"

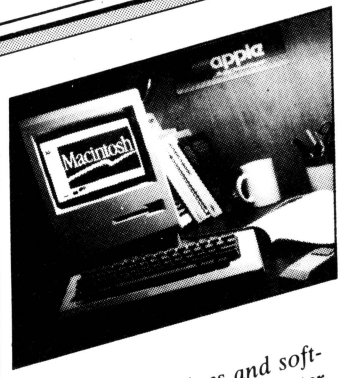
Into computing? This is your magazine! Just look what's in it for you!

NEWS



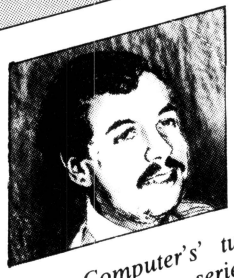
'Your Computer' brings you all that's interesting, innovative and inventive in the microcomputing world — news of products, plans and politics to keep you up-to-date with what's going on in this fast-moving industry.

REVIEWS



The latest machines and software from all the computer manufacturers are reviewed each month in 'Your Computer'. Keep up-to-date with what's available, and use our reviews to help you assess which products are most useful for you.

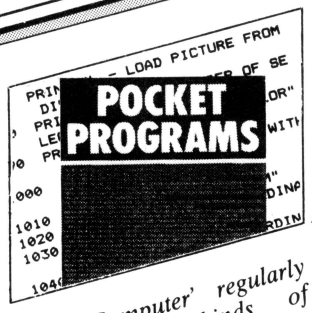
TUTORIALS



Les Bell

'Your Computer's' tutorials include regular series on such popular subjects as the BASIC programming language and dBase II, probably the biggest-selling database program of them all. Written by such well-known industry experts as Les Bell, they're an invaluable aid to learning how to make computers work for you.

PROGRAMS

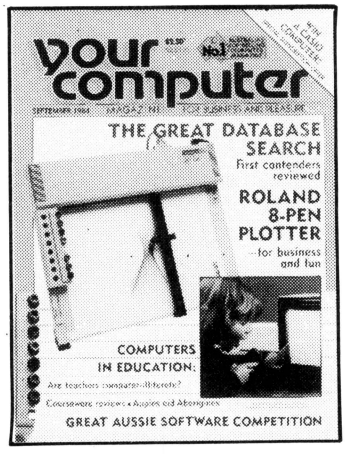


'Your Computer' regularly publishes all kinds of programs written both by professional programmers and readers, and ranging from games to business uses, utilities to additions and alterations to well-known programs. All kinds of computer users and enthusiasts will find programs to suit them in our pages!

No.1 AUSTRALIA'S TOP-SELLING COMPUTER MAGAZINE
AUDITED SALES

Your Computer

is a magazine for all computer users and enthusiasts — there's something for everyone in our pages. We cover games and utility programs for hobbyists, and every second month we publish 'Your Business Computer' as a separate section for business computer users. 'Your Computer' offers topical features on all aspects of the computing world, written in understandable everyday language, along with reviews of new products to help you find your way round the fast-growing market, and regular columns giving specific advice on all the most popular brands of microcomputer. If computers play any part in your life at all, don't miss your monthly issue of 'Your Computer' — without it, you just won't know what's going on!



Available at your newsagent now!

```
2000 PRINTCHR$(147):PRINT:PRINT"NO. OF VERTICES"
2010 INPUT"(MAXIMUM 48)";NV
2020 IFNV<1ORNV>48THEN2000
2030 FORI=1TONV
2040 PRINTCHR$(147):PRINT"VERTEX";I
2050 PRINT"(CO-ORDINATE RANGE -50 TO 50)
2060 INPUT"X=";X(I)
2065 IFX(I)<-50ORX(I)>50THEN2040
2070 INPUT"Y=";Y(I)
2075 IFY(I)<-50ORY(I)>50THEN2040
2080 INPUT"Z=";Z(I)
2085 IFZ(I)<-50ORZ(I)>50THEN2040
2090 NEXT
2100 PRINTCHR$(147):PRINT:PRINT"NO. OF EDGE INDICES"
2110 INPUT"(MAXIMUM 96)";NE
2115 IFNE<2ORNE>96THEN2100
2120 FORI=1TONE
2130 PRINTCHR$(147):PRINT"INDEX";I
2140 INPUT"(NEGATIVE TO SUPRESS DRAWING)";E(I)
2150 NEXT
2160 RETURN
3000 PRINTCHR$(147):PRINT:PRINT"INSERT BLANK DATA TAPE AND
     PRESS RETURN"
3010 GETM$:IFM$<>CHR$(13)THEN3010
3020 OPEN1,1,1,"SHAPE"
3030 PRINT#1,NV
3040 FORI=1TONV
3050 PRINT#1,X(I)
3060 PRINT#1,Y(I)
3070 PRINT#1,Z(I)
3080 NEXT
3090 PRINT#1,NE
3100 FORI=1TONE
3110 PRINT#1,E(I)
3120 NEXT
3130 CLOSE1
3140 PRINTCHR$(147):PRINT"SAVING COMPLETE"
3150 GOTO350


5 PRINTCHR$(147):PRINT:PRINTTAB(3)"[R]OUTINES TAKE 109 SECONDS TO
  LOAD":PRINT
10 DIMSP(11):DIMCH(11)
100 FORX=1TO11
110 READSP(X):READCH(X)
120 NEXT
130 FORX=1TO11
140 GOSUB1000
150 IFSP=SP(X)ANDCH=CH(X)THEN170
160 PRINT"ERROR IN BLOCK";X:END
170 CH=0:PRINT"BLOCK";X;"LOADED":NEXT
180 PRINT:PRINTTAB(10)"[R]OUTINES LOADED OK":END
1000 READSP
1010 READA$:IFA$="*"THENRETURN
1020 L=ASC(LEFT$(A$,1))-48:R=ASC(RIGHT$(A$,1))-48
1030 PC=(L+7*(L>11))*16+R+7*(R>11)
1040 POKESP,PC:CH=CH+PC
1050 SP=SP+1:GOTO1010
4000 DATA 17376,15061,17536,23647,17696,19175,17856,19595,18016,18689
4010 DATA 18176,19006,18336,17639,18496,16283,18656,13524,18816,15100
4020 DATA 18979,17701
4999 DATA 17216
```

```
5000 DATA EC,31,43,F0,1E,38,ED,2F
5010 DATA 43,48,98,ED,30,43,10,1C
5020 DATA A8,68,49,FF,18,69,01,48
5030 DATA 98,49,FF,69,00,AE,31,43
5040 DATA 4C,6C,43,18,6D,2F,43,48
5050 DATA 98,6D,30,43,A8,68,60,EA
5060 DATA 8D,2B,43,8D,2C,43,A2,11
5070 DATA 18,6E,2C,43,6E,2B,43,6E
5080 DATA 2A,43,6E,29,43,90,13,18
5090 DATA AD,27,43,6D,2B,43,8D,2B
5100 DATA 43,AD,28,43,6D,2C,43,8D
5110 DATA 2C,43,CA,D0,DC,60,EA,EA
5120 DATA A9,00,8D,37,43,8D,2D,43
5130 DATA 8D,2E,43,AD,22,43,D0,07
5140 DATA 8D,2F,43,8D,30,43,60,A2
5150 DATA 10,2E,2F,43,2E,30,43,2E
5160 DATA 2D,43,2E,2E,43,38,AD,2D
5170 DATA 43,ED,22,43,A8,AD,2E,43
5180 DATA ED,37,43,90,06,8C,2D,43
5190 DATA 8D,2E,43,CA,D0,DB,2E,2F,*
5199 DATA 17376
5200 DATA 43,2E,30,43,60,EA,EA,EA
5210 DATA BD,A6,42,D0,12,BD,A0,42
5220 DATA C9,2D,D0,07,DE,A6,42,DE
5230 DATA A0,42,60,FE,A0,42,60,BD
5240 DATA A0,42,D0,13,FE,A6,42,FE
5250 DATA A0,42,BC,AC,42,F0,02,C8
5260 DATA C8,88,98,9D,AC,42,60,DE
5270 DATA A0,42,60,EA,EA,EA,EA,EA
5280 DATA 20,E8,43,CA,CA,CA,10,F8
5290 DATA A9,01,8D,34,43,60,EA,EA
5300 DATA BD,A6,42,D0,1C,BD,A0,42
5310 DATA D0,13,DE,A6,42,FE,A0,42
5320 DATA BC,AC,42,F0,02,C8,C8,88
5330 DATA 98,9D,AC,42,60,DE,A0,42
5340 DATA 60,BD,A0,42,C9,2D,D0,07
5350 DATA FE,A6,42,DE,A0,42,60,FE
5360 DATA A0,42,60,EA,EA,EA,EA,EA
5370 DATA EA,20,30,44,CA,CA,CA,10
5380 DATA F8,A9,01,8D,34,43,60,EA
5390 DATA EA,EA,EA,EA,EA,EA,EA,EA,*
5399 DATA 17536
5400 DATA A2,02,A9,2D,9D,A3,42,CA
5410 DATA 10,FA,A2,02,A9,FF,9D,A9
5420 DATA 42,9D,AC,42,9D,AF,42,CA
5430 DATA 10,F4,A2,02,A9,00,9D,A0
5440 DATA 42,9D,A6,42,CA,10,F7,8D
5450 DATA AE,42,8D,B1,42,8D,32,43
5460 DATA 8D,33,43,20,B0,C1,A2,05
5470 DATA BC,A0,42,B9,00,40,9D,B2
5480 DATA 42,CA,10,F4,AD,B4,42,8D
5490 DATA 27,43,AD,B5,42,8D,29,43
5500 DATA A9,00,8D,28,43,8D,2A,43
5510 DATA 20,70,43,AD,29,43,8D,BA
5520 DATA 42,AD,2A,43,8D,C3,42,A2
5530 DATA 00,AD,AE,42,CD,AF,42,F0
5540 DATA 01,CA,8E,CC,42,AD,B6,42
5550 DATA 8D,29,43,A9,00,8D,2A,43
5560 DATA 20,70,43,AD,29,43,8D,DC
5570 DATA 42,8D,D5,42,AD,2A,43,8D
5580 DATA E0,42,8D,D9,42,A2,FF,AD
5590 DATA AE,42,CD,B0,42,F0,01,E8,*
5599 DATA 17696
5600 DATA 8E,E8,42,8E,E5,42,AD,B3
5610 DATA 42,8D,29,43,A9,00,8D,2A
5620 DATA 43,20,70,43,AD,29,43,8D
5630 DATA DB,42,8D,D6,42,AD,2A,43
5640 DATA 8D,DF,42,8D,DA,42,A2,00
5650 DATA A0,FF,AD,AE,42,CD,AD,42
5660 DATA F0,02,C8,CA,8E,E6,42,8C
5670 DATA E7,42,AD,B3,42,8D,27,43
5680 DATA AD,B7,42,8D,29,43,A9,00
5690 DATA 8D,28,43,8D,2A,43,20,70
5700 DATA 43,AD,29,43,8D,D4,42,8D
5710 DATA DD,42,AD,2A,43,8D,D8,42
5720 DATA 8D,E1,42,A2,FF,AD,AD,42
5730 DATA CD,B1,42,F0,01,E8,8E,E4
5740 DATA 42,8E,E9,42,AD,B5,42,8D
5750 DATA 29,43,A9,00,8D,2A,43,20
5760 DATA 70,43,AD,29,43,8D,BB,42
5770 DATA AD,2A,43,8D,C4,42,A2,00
5780 DATA AD,AD,42,CD,AF,42,F0,01
5790 DATA CA,8E,CD,42,AD,B6,42,8D,*
5799 DATA 17856
5800 DATA 27,43,AD,B7,42,8D,29,43
5810 DATA A9,00,8D,28,43,8D,2A,43
5820 DATA 20,70,43,AD,29,43,8D,D3
5830 DATA 42,8D,DE,42,AD,2A,43,8D
5840 DATA D7,42,8D,E2,42,A2,00,A0
5850 DATA FF,AD,B0,42,CD,B1,42,F0
5860 DATA 02,CA,C8,8E,E3,42,8C,EA
5870 DATA 42,AD,B5,42,8D,29,43,A9
5880 DATA 00,8D,2A,43,20,70,43,AD
5890 DATA 29,43,8D,BC,42,AD,2A,43
5900 DATA 8D,C5,42,A2,00,AD,B0,42
5910 DATA CD,AF,42,F0,01,CA,8E,CE
5920 DATA 42,AD,B5,42,8D,27,43,AD
5930 DATA B7,42,8D,29,43,A9,00,8D
5940 DATA 28,43,8D,2A,43,20,70,43
5950 DATA AD,29,43,8D,C0,42,AD,2A
5960 DATA 43,8D,C9,42,A2,00,AD,AF
5970 DATA 42,CD,B1,42,F0,01,CA,8E
5980 DATA D2,42,AD,B2,42,8D,C6,42
5990 DATA A9,00,8D,BD,42,AD,AC,42,*
5999 DATA 18016
6000 DATA 8D,CF,42,A2,03,18,7E,C3
6010 DATA 42,7E,BA,42,18,7E,D7,42
6020 DATA 7E,D3,42,18,7E,DF,42,7E
6030 DATA DB,42,CA,10,E8,18,6E,C9
6040 DATA 42,6E,C0,42,A0,03,8C,35
6050 DATA 43,AD,B2,42,8D,28,43,A9
6060 DATA 00,8D,27,43,AC,35,43,BE
6070 DATA DB,42,8E,29,43,BE,DF,42
6080 DATA 8E,2A,43,20,70,43,AC,35
6090 DATA 43,A2,00,AD,AC,42,D9,E7
7000 DATA 42,F0,01,CA,B9,E3,42,8D
7010 DATA 31,43,B9,D3,42,8D,2F,43
7020 DATA B9,D7,42,8D,30,43,AD,2B
7030 DATA 43,AC,2C,43,20,40,43,48
7040 DATA 98,AC,35,43,99,EF,42,68
7050 DATA 99,EB,42,8A,99,F3,42,CE
7060 DATA 35,43,10,AB,A0,01,B9,EB
7070 DATA 42,99,B8,42,B9,EF,42,99
7080 DATA C1,42,B9,F3,42,99,CA,42
7090 DATA 88,10,EB,A0,01,B9,ED,42,*
7099 DATA 18176
7100 DATA 99,BE,42,B9,F1,42,99,C7
7110 DATA 42,B9,F5,42,99,D0,42,88
7120 DATA 10,EB,AC,2E,40,8C,35,43
7130 DATA B9,5F,40,8D,31,43,B9,2F
7140 DATA 40,8D,2F,43,AE,CD,42,A9
7150 DATA 00,8D,30,43,A8,AD,C4,42
7160 DATA 20,40,43,8E,00,43,8D,FD
7170 DATA 42,AC,35,43,B9,BF,40,8D
7180 DATA 31,43,B9,8F,40,8D,2F,43
7190 DATA AE,CE,42,A9,00,8D,30,43
7200 DATA A8,AD,C5,42,20,40,43,8E
7210 DATA 01,43,8D,FE,42,AC,35,43
7220 DATA B9,1F,41,8D,31,43,B9,EF
7230 DATA 40,8D,2F,43,AE,CF,42,A9
7240 DATA 00,8D,30,43,A8,AD,C6,42
7250 DATA 20,40,43,8E,02,43,8D,FF
7260 DATA 42,A0,02,8C,36,43,B9,B8
7270 DATA 42,8D,29,43,B9,C1,42,8D
7280 DATA 2A,43,B9,FD,42,8D,28,43
7290 DATA A9,00,8D,27,43,20,70,43,*
7299 DATA 18336
7300 DATA AD,2B,43,AC,36,43,99,03
7310 DATA 43,AD,2C,43,99,06,43,A2
7320 DATA 00,B9,CA,42,D9,00,43,F0
7330 DATA 01,CA,8A,99,09,43,CE,36
7340 DATA 43,AC,36,43,10,C0,A0,02
7350 DATA 8C,36,43,B9,BB,42,8D,29
7360 DATA 43,B9,C4,42,8D,2A,43,B9
7370 DATA FD,42,8D,28,43,A9,00,8D
7380 DATA 27,43,20,70,43,AD,2B,43
7390 DATA AC,36,43,99,0C,43,AD,2C
7400 DATA 43,99,0F,43,A2,00,B9,CD
7410 DATA 42,D9,00,43,F0,01,CA,8A
7420 DATA 99,12,43,CE,36,43,AC,36
7430 DATA 43,10,C0,A0,02,8C,36,43
7440 DATA B9,BE,42,8D,29,43,B9,C7
7450 DATA 42,8D,2A,43,B9,FD,42,8D
7460 DATA 28,43,A9,00,8D,27,43,20
7470 DATA 70,43,AD,2B,43,AC,36,43
7480 DATA 99,15,43,AD,2C,43,99,18
7490 DATA 43,A2,00,B9,D0,42,D9,00,*
7499 DATA 18496
7500 DATA 43,F0,01,CA,8A,99,1B,43
7510 DATA CE,36,43,AC,36,43,10,C0
7520 DATA AD,03,43,8D,2F,43,AD,06
7530 DATA 43,8D,30,43,AD,09,43,8D
7540 DATA 31,43,AD,04,43,AC,07,43
7550 DATA AE,0A,43,20,40,43,48,AD
7560 DATA 05,43,8D,2F,43,AD,08,43
7570 DATA 8D,30,43,AD,0B,43,8D,31
7580 DATA 43,68,20,40,43,8D,1E,43
7590 DATA 8C,1F,43,8E,20,43,AD,0C
7600 DATA 43,8D,2F,43,AD,0F,43,8D
7610 DATA 30,43,AD,12,43,8D,31,43
7620 DATA AD,0D,43,AC,10,43,AE,13
7630 DATA 43,20,40,43,48,AD,0E,43
7640 DATA 8D,2F,43,AD,11,43,8D,30
7650 DATA 43,AD,14,43,8D,31,43,68
7660 DATA 20,40,43,8D,21,43,8C,22
7670 DATA 43,8E,23,43,AD,15,43,8D
7680 DATA 2F,43,AD,18,43,8D,30,43
7690 DATA AD,1B,43,8D,31,43,AD,16,*
7699 DATA 18656
7700 DATA 43,AC,19,43,AE,1C,43,20
7710 DATA 40,43,48,AD,17,43,8D,2F
7720 DATA 43,AD,1A,43,8D,30,43,AD
7730 DATA 1D,43,8D,31,43,68,20,40
7740 DATA 43,8D,24,43,8C,25,43,8E
7750 DATA 26,43,AD,1E,43,8D,2F,43
7760 DATA AD,1F,43,8D,30,43,20,A0
7770 DATA 43,AD,20,43,A2,00,CD,23
7780 DATA 43,F0,01,CA,8E,31,43,A9
7790 DATA 9F,A0,00,A2,00,20,40,43
7800 DATA 48,98,AC,35,43,99,3F,42
7810 DATA 68,99,0F,42,AD,24,43,8D
7820 DATA 2F,43,AD,25,43,8D,30,43
7830 DATA 20,A0,43,AD,26,43,A2,FF
7840 DATA CD,23,43,F0,01,E8,8E,31
7850 DATA 43,A9,63,A0,00,A2,00,20
7860 DATA 40,43,AC,35,43,99,6F,42
7870 DATA CE,35,43,F0,06,AC,35,43
7880 DATA 4C,18,47,20,CB,C1,A9,01
7890 DATA 85,FD,AD,2F,40,8D,35,43,*
7899 DATA 18816
7900 DATA AC,35,43,B9,AF,41,D0,19
7910 DATA B9,4F,41,A8,B9,0F,42,8D
7920 DATA AB,C1,B9,3F,42,8D,AC,C1
7930 DATA B9,6F,42,8D,AA,C1,20,7F
7940 DATA C2,AC,35,43,B9,4F,41,A8
7950 DATA B9,0F,42,8D,AE,C1,B9,3F
7960 DATA 42,8D,AF,C1,B9,6F,42,8D
7970 DATA AD,C1,CE,35,43,D0,C1,A9
7980 DATA 00,8D,34,43,AD,00,DC,A2
7990 DATA 03,4A,B0,05,48,20,20,44
8000 DATA 68,4A,B0,05,48,20,69,44
8010 DATA 68,A2,04,4A,B0,05,48,20
8020 DATA 20,44,68,4A,B0,05,48,20
8030 DATA 69,44,68,4A,B0,21,AD,32
8040 DATA 43,D0,0B,AC,33,43,F0,02
8050 DATA C8,C8,88,8C,33,43,A2,05
8060 DATA AD,33,43,D0,05,20,20,44
8070 DATA D0,07,20,69,44,D0,02,A9
8080 DATA 00,8D,32,43,A5,C5,C9,03
8090 DATA F0,08,AD,34,43,F0,A5,4C
8100 DATA B6,44,60,*

READY.
```

■

START MATHS

Start Maths is an addition and subtraction drilling program for the Commodore 64 suitable for ages 6 to 12. It first asks for the level of difficulty; enter '1' for easy to '5' for hard. You will then be given five sums involving addition and five involving subtraction.

After the ten sums have been completed you will be asked if you want any more sums. If you require more press 'y', if not then press 'n' to be returned to BASIC.

Simon Jones
Holder ACT


```
5 POKE53280,14:POKE53281,14
10 PRINT"♥";
20 PRINT"QQR          ***START MATHS***          ■"
30 IFSS=0THENGOSUB800:SS=1
40 PRINT"QQQQQQQIIIIYOU WILL BE GIVEN 10 SUMS INVOLVING  ADDITION ";
50 PRINT"AND SUBTRACTION TO SOLVE."
60 PRINT:PRINT:PRINT"LEVEL OF SUMS (1-EASY..5-HARD) R?■"
70 GET L$:IFL$>"0"ANDL$<"6"THEN90
80 GOTO70
90 L=VAL(L$)
100 INPUT"QQQWHAT IS YOUR NAME";NA$
110 PRINT"QQQR          PRESS ANY KEY TO START          ■"
120 POKE198,0:WAIT198,1:POKE198,0
130 FORSM=1TO10
140 PRINT"♥";
150 W=1:A1=1
160 N1=INT(RND(0)*10↑L)
170 N2=INT(RND(0)*10↑L)
180 IFN1<N2THEN160
190 IFSM<6THENS$="+":AN=N1+N2
200 IFSM>5THENS$="-":AN=N1-N2
210 N1$=STR$(N1):N2$=STR$(N2)
220 L1=LEN(N1$)-1:L2=LEN(N2$)-1
230 PRINT:PRINT:PRINT:PRINTTAB(15-LEN(N1$));N1$
240 IFS$="+"THENPRINT:PRINTTAB(7)"■+■";
```

```
250 IFS$="-"THENPRINT:PRINTTAB(7)"CE-E";
260 PRINT"SQQQQ";
270 PRINTSPC(15-LEN(N2$));N2$
280 PRINT"      ─────────────────Q"
290 PRINT"      ─────────────────"
300 PRINT"SQQQQQQ";:PRINTSPC(15-A1)"?"
310 PRINT"SQQQQQQ";:PRINTSPC(15-A1)"?"
320 POKE198,0
330 GET A$:IFA$=""THEN330
340 PRINT"QQQQQQQQQQQQQQQQQQQQ                    ":PRINT"               "
350 IFA$>="0"ANDA$<":"THEN370
360 GOTO330
370 PRINT"SQQQQQQ";:PRINTSPC(15-A1);A$
380 IFA$=MID$(STR$(AN),LEN(STR$(AN))-A1+1,1)THENGOTO450
390 IFW<3THENPRINT"SQQQQQQQQQQQQQQQQQQQRWRONG!!":PRINT"RTRY AGAIN."
    :W=W+1:GOSUB570 :GOTO300
400 GOSUB570:PRINT"SQQQQQQQQQQQQQQQQQQQRTHE ANSWER":PRINT"RIS -■":W=1
410 PRINT"SQQQQQQ";SPC(15-A1);
420 PRINTMID$(STR$(AN),LEN(STR$(AN))-A1+1,1)
430 IFA1=LEN(STR$(AN))-1THENFORI=1TO800:NEXT:NEXTSM:GOTO510
440 A1=A1+1:GOTO300
450 REM CORRECT ANSWER
460 IFA1=LEN(STR$(AN))-1THENGOSUB640:SC=SC+1:GOSUB750:NEXTSM:GOTO510
470 A1=A1+1
480 PRINT"SQQQQQQQQQQQQQQQQQQQRCORRECT!!■     ":W=1
490 GOSUB640:GOTO300
500 FORI=1TO1000:NEXTI:NEXTSM
510 PRINT"♥";
520 PRINT"SQQQQQQQQQDO YOU WANT SOME MORE SUMS RY■/RN■ R?■";
530 GETC$:IFC$=""THEN530
540 IFC$="N"THEN1120
550 IFC$="Y"THEN10
560 GOTO530
570 V=54272
580 FORI=0TO24:POKEV+I,0:NEXTI
590 POKEV+1,6:POKEV,206
600 POKEV+5,50:POKEV+6,255
610 POKEV+4,33:POKEV+24,15
620 FORI=1TO300:NEXTI:POKEV+1,0:POKEV,0
630 RETURN
640 V=54272
650 FORI=0TO24:POKEV+I,0:NEXTI
660 POKEV+1,91:POKEV,140
670 POKEV+5,50:POKEV+6,255
680 POKEV+4,17:POKEV+24,15
690 FORI=1TO150:NEXTI
700 POKEV+1,68:POKEV,149
710 FORI=1TO150:NEXTI:POKEV+1,0:POKEV,0
720 RETURN
730 S=54272
740 FORI=0TO24:POKEV+I,0:NEXTI
750 PRINT"♥";
760 PRINT"QQQQQQQQQQQQQQQ              RGREAT WORK ";NA$;"!!!■"
770 FORI=1TO255STEP15:POKE53280,I:POKE53281,I:FORZ=1TO80:NEXTZ:NEXTI
780 POKE53280,14:POKE53281,14
790 RETURN
800 S=54272
810 FORI=0TO24:POKES+I,0:NEXTI
820 POKES+3,7:POKES+2,75
830 POKES+24,0
840 FORI=1TO20
850 READ H,L,D
860 POKES+1,H:POKES,L
870 POKES+5,64:POKES+6,75
880 POKES+4,65:POKES+24,15
890 FORZ=1TOD*8:NEXTZ
900 NEXTI
910 FORI=0TO24:POKES+I,0:NEXTI:RETURN
920 DATA 9,159,30:REM D
930 DATA 10,205,20:REM E
940 DATA 11,114,20:REM F
950 DATA 12,216,20:REM G
960 DATA 8,147,20:REM C
970 DATA10,205,20:REM E
980 DATA 9,159,20:REM D
990 DATA 8,23,20:REM B
1000 DATA 11,114,20:REM F
1010 DATA 10,205,20:REM E
1020 DATA 9,159,20:REM D
1030 DATA 8,147,20:REM C
1040 DATA 12,216,20:REM G
1050 DATA 11,114,20:REM F
1060 DATA 9,159,20:REM D
1070 DATA 10,205,20:REM E
1080 DATA 8,147,20:REM C
1090 DATA 9,159,20:REM D
1100 DATA 8,23,20:REM B
1110 DATA 8,147,40:REM C
1120 END
```

HI-RES

What's needed for the Commodore is a package of cheap graphics routines which will enable high resolution graphics mode and make it easy to plot dots, lines, circles et cetera – so here they are.

The HIRES package of machine code subroutines is loaded into the Commodore 64's memory from $C000 (49152) to $C901 (51457), an area of memory which is not used by BASIC. The routines are accessible from BASIC by simple POKE and SYS commands. They allow setting up of bit mapped (high resolution) graphics mode, clearing the bit mapped screen, setting the colour of the foreground and background, and the plotting of dots, lines, circles and ellipses. With these routines in memory, you can write your own BASIC programs using the subroutines described below.

The subroutines which make up the package may be found at the following memory locations:

Subroutine SETUP $C1B0 (49584) to $C1CA

Subroutine CLRSCN $C1CB (49611) to $C203

Subroutine SETCOLR $C204 (49668) to $C22E

Subroutine DOT $C22F (49711) to $C27E

Subroutine LINE $C27F (49791) to $C475

Subroutine CIRCLE $C721 (50977) to $C901

The remaining memory space is reserved for variables used by the routines.

To initialize bit mapped mode:

SYS 49584 (SETUP)

This routine calls two other routines:

a) Subroutine SLRSCN at 49611 which clears the bit map. This routine can be used by itself to clear the screen at any time by (SYS 49611).

b) Subroutine SETCOLR at 49668 which sets the colour of each area of the bit map. The colour defaults to 3 (black on cyan), but may be altered by POKEing 49681 ($C211) with the number shown below, before calling either SETCOLR or SETUP.

POKE 49681,N where N = SC + FO*16

SC is the required screen colour and FO is the required foreground (dot) colour. In both cases, zero = black, one = white, two = red, etc ...

To plot a dot at any location (X,Y) on the screen, where X = 0 to 319 and Y = 0 to 199, X must first be split into high and low bytes, so a complete routine would be:

```
SYS 49584 (SETUP)
POKE 253,1 (SWITCH TOGGLE ON)
HX=INT(X/256):LX=X-256*HX (SPLIT X)
POKE 49581,Y:POKE 49582,LX:POKE 49583,HX (POKE X,Y)
SYS 49711 (DOT)
```

The last three lines can obviously be written as a BASIC subroutine and called whenever required.

To plot a line on the screen the switch (2543) must be set as above, then the starting (X1,Y1) and finishing (X2,Y2) coordinates of the line must be POKEd. Again each X coordinate must be split into two bytes.

```
40 REM     HIRES ROUTINES
50 REM
60 REM  MIKE GRIFFITHS - 1983
70 REM
80 REM
90 PRINTCHR$(147):PRINT:PRINTTAB(4)"ROUTINES TAKE 104 SECONDS TO LOAD"
100 LS=49152:HS=49360:SS=8192:C=49352
110 FORY=0TO24
120 FORX=0TO7
130 HB=INT((SS+X)/256):LB=SS+X-HB*256
140 POKELS,LB:POKEHS,HB
150 LS=LS+1:HS=HS+1
160 NEXT:SS=SS+320:NEXT
170 BI=128
180 FORX=0TO7
190 POKE49568+X,BI
200 BI=BI/2:NEXT
210 FORX=0TO255
220 SX=SIN(.*X/512)*65536
230 HB=INT(SX/256):LB=SX-256*HB
240 POKE50304+X,LB:POKE50560+X,HB
250 NEXT
260 POKEC+4,0:POKEC+5,0:POKEC+6,100:POKEC+7,0
1000 GOSUB1500
1005 PRINT
1010 IFCH=99583ANDSP=50294THEN1060
1050 PRINT"DATA CHECKSUM ERROR IN BLOCK ONE":END
1060 CH=0:GOSUB1500
1070 PRINT
1080 IFCH=95743ANDSP=51458THENPRINT"OK TO LOAD APPLICATIONS PROGRAM":END
1090 PRINT"DATA CHECKSUM ERROR IN BLOCK TWO":END
1500 READ SP
1520 READA$:IFA$="*"THENRETURN
1530 L=ASC(LEFT$(A$,1))-48:R=ASC(RIGHT$(A$,1))-48
1540 PC=(L+7*(L>11))*16+R+7*(R>11)
1550 POKESP,PC:CH=CH+PC
```

SYS 49584 (SETUP)
POKE 253,1 (SWITCH)
POKE 49581,Y1:POKE 49578,Y2 (POKE Y1,Y2)
HX=INT(X1/256):LX=X1-256*HX (SPLIT X1)
POKE 49582, LX:POKE49583,HX (POKE X1)
HX=INT(X2/256):LKX=X2-256*HX (SPLIT X2)
POKE 49579,LX:POKE 49580,HX (POKE X2)
SYS 49791 (LINE)

Again, the last six lines can be set up as a subroutine in BASIC to plot any line (X1,Y1) to (X2,Y2). It should be noted that SETUP may only need to be called once in a particular program, and the switch remains set unless SETUP, CLRSCN or POKE 253,0 is used. These two lines are, therefore, excluded from the subroutine.

To plot a circle, ellipse or arc a large number of parameters can be specified to describe the required curve. The routine below includes all options; note that a variable C has been used to simplify the routine.

```
100 C=49352                              (SET C)

200 SYS 49584                            (SETUP)

300 POKE 253,1                           (SWITCH)

500 XH=INT(XC/256):XL=XC-256*XH          (SPLIT XC)

600 POKE C,XL:POKE C+1,XH:POKE C+2,YC    (POKE XC,YC)

700 POKE C+3,RX                          (POKE RX)

800 POKE C+4,RY                          (POKE RY)

900 POKE C+5,IA:POKE C+6,FA              (POKE IA,FA)

1000 POKE C+7,GW                         (POKE GW)

1010 SYS 50977                           (CIRCLE)
```

(XC,YC) are the coordinates of the centre of the circle or ellipse, with XC again having to be split into two bytes. RX is the radius in the X direction and RY is the radius in the Y direction (up to 255 each). IA and FA are the initial angle and final angle respectively, where a full circle has 100 degrees. Curves are always drawn clockwise, so 0 = 'east', 25 = 'south', 50 = 'west', and 75 = 'north'.

GW is a variable which can be used to produce dotted curves. GW = 0 or 1 will produce solid curves up to radii of 160. Setting GW to a higher number (for example 10) will produce a dotted curve (GW = gap width between dots).

The above subroutine can be simplified considerably if all options are not required (default conditions will apply). If only solid curves are required, leave out line 1000 (GW defaults to zero). If complete curves are required rather than arcs, leave out line 900 (IA defaults to zero, FA defaults to 100). IF circles are required rather than ellipses, leave out line 800 (RY defaults to = RX).

When the routines are loaded into memory, they leave all of the BASIC memory area untouched. However the bit mapped screen, starting at 8192 ($2000) is in the middle of the BASIC program area. Thus a program plus variables of more than 8K will 'crash into' the high resolution screen memory area. This can be avoided by setting the bottom of BASIC above the hires screen area to 16384 ($4000), before loading a subsequent large applications program, leaving about 24K RAM available for BASIC. This is a system limitation, but not a serious one, since most applications programs will not approach this size.

The routines are loaded by the BASIC program listed in this article. Once loaded the BASIC loader program can be cleared, and the machine code remains in memory until you turn the power off.

The routines can be called from machine code by using JSR and the appropriate hex. address. Machine code programmers should note, however, that locations 251 ($FB), 252 ($FC), and 253 ($FD) in zero page memory are used by the routines, and must therefore be avoided by applications programs.

If you have a machine code monitor you can speed up subsequent loading of the package

```
1570 SP=SP+1:GOTO1520
2000 DATA 49584
2050 DATA 78,20,CB,C1,78,A9,08,0D,18,D0,8D,18,D0,20,04,C2
2060 DATA 78,A9,20,0D,11,D0,8D,11,D0,58,60
2110 DATA 78,A9,00,85,FD,85,FB,A9,20,85,FC,A0,08,88,A9,00,91,FB,C0,00,F0,03
2120 DATA 4C,D8,C1,A9,08,18,65,FB,85,FB,90,03,E6,FC,18,A5,FC,C9,3F,F0,03,4C
2130 DATA D6,C1,A5,FB,C9,40,F0,03,4C,D6,C1,58,60
2210 DATA 78,A9,00,85,FB,A9,04,85,FC,AA,A0,FA,A9,03,88,91,FB,F0,03
2220 DATA 4C,12,C2,18,A9,FA,65,FB,85,FB,A9,00,65,FC,85,FC,CA,F0,03,4C,0E,C2
2230 DATA 58,60
2310 DATA 78,AC,AD,C1,C0,C7,B0,46,B9,00,C0,85,FB,B9,D0,C0,85,FC,AD,AF,C1
2315 DATA F0,0D,C9,01,D0,33
2320 DATA AA,AD,AE,C1,C9,40,B0,2B,8A,18,65,FC,85,FC,AD,AE,C1,AA,4A,4A,4A
2330 DATA 0A,0A,0A,A8,8A,29,07,AA,A5,FD,F0,09,BD,A0,C1,11,FB,91,FB,58,60
2340 DATA BD,A0,C1,49,FF,31,FB,91,FB,58,60
2400 DATA 78,20,2F,C2,78,AD,AB,C1,38,ED,AE,C1,A8,AD,AC,C1,ED,AF,C1,B0,15,A2,02
2410 DATA 8E,A8,C1,AA,98,49,FF,18,69,01,A8,8A,49,FF,69,00,4C,BB,C2,C9,00
2420 DATA D0,09,C0,00,D0,05,A2,00,4C,B8,C2,A2,01,8E,A8,C1,8C,9A,C1,8D,9B,C1
2430 DATA AD,AA,C1,38,ED,AD,C1,90,0C,8D,98,C1,C9,00,F0,0F,A9,01
2440 DATA 4C,E0,C2,49,FF,18,69,01,8D,98,C1,A9,02,8D,A9,C1,C9,00,D0,03,4C,11,C4
2450 DATA AD,A8,C1,D0,03,4C,EC,C3,A9,00,8D,9C,C1,8D,9D,C1
2460 DATA AD,9B,C1,F0,03,4C,6C,C3,AD,9A,C1,38,ED,98,C1,90,03,4C,6C,C3,AC,A9,C1
2470 DATA AE,A8,C1
2480 DATA C0,01,D0,06,EE,AD,C1,4C,21,C3,CE,AD,C1,20,5C,C4,48,AD,9D,C1,ED,99,C1
2490 DATA 30,2E,8D,9D,C1,68,8D,9C,C1,48,E0,01,D0,14,AD,AE,C1,18,69,01,8D,AE,C1
2500 DATA AD,AF,C1,69,00,8D,AF,C1,4C,5B,C3,AD,AE,C1,38,E9,01,8D,AE,C1
2505 DATA B0,03,CE,AF,C1
2510 DATA 68,20,2F,C2,AD,AA,C1,CD,AD,C1,F0,03,4C,0E,C3,58,60,AD,98,C1,48
2520 DATA AD,9A,C1,8D,98,C1,AD,9B,C1,8D,99,C1,68,8D,9A,C1,AC,A9,C1,AE,A8,C1
2530 DATA E0,01,D0,14,AD,AE,C1,18,69,01,8D,AE,C1,AD,AF,C1,69,00
2540 DATA 8D,AF,C1,4C,B1,C3,AD,AE,C1,38,E9,01,8D,AE,C1,B0,08,AD,AF,C1,F0,03
2550 DATA CE,AF,C1,20,5C,C4,48,AD,9D,C1,ED,99,C1,30,15,8D,9D,C1,68,8D,9C,C1
2560 DATA 48,C0,01,D0,06,EE,AD,C1,4C,D2,C3,CE,AD,C1,68,20,2F,C2,78,AD,AC,C1
2570 DATA CD,AF,C1,D0,0A,AD,AB,C1,CD,AE,C1,D0,02,58,60,4C,80,C3,AC,A9,C1,D0,02
2575 DATA 58,60,C0,01,F0,06,CE,AD,C1,4C,00,C4,EE,AD,C1,20,2F,C2,78,AD,AD,C1
```

by saving it directly in machine code form as below:

With your monitor in memory, load and run the BASIC program. Enter the monitor and save HIRES by –

S"HIRESMC",01,C000,C902
S"0:HIRESMC",08,C000,C902

You can then use LOAD"HIRESMC",1,1 or LOAD"HIRESMC",8,1 to load the package directly into the correct place in memory.

M. Griffiths
Lindisfarne NSW

HI-RES

```
2580 DATA CD,AA,C1,F0,03,4C,EC,C3,58,60,AE,A8,C1,D0,02,58,60,E0,01,D0,14
2590 DATA AD,AE,C1,18,69,01,8D,AE,C1,AD,AF,C1,69,00,8D,AF,C1,4C,43,C4,AD,AE,C1
2600 DATA 38,E9,01,8D,AE,C1,B0,08,AD,AF,C1,F0,03,CE,AF,C1,20,2F,C2,78
2610 DATA AD,AF,C1,CD,AC,C1,D0,0A,AD,AE,C1,CD,AB,C1,D0,02,58,60,4C,11,C4
2620 DATA AD,9C,C1,18,6D,9A,C1,8D,9C,C1,AD,9D,C1,69,00,8D,9D,C1,AD,9C,C1
2630 DATA 38,ED,98,C1,60,*
2700 DATA 50832
2710 DATA 8D,84,C6,8D,85,C6,A2,11,18,6E,85,C6,6E,84,C6,6E,83,C6,6E,82,C6,90,13
2720 DATA 18,AD,80,C6,6D,84,C6,8D,84,C6,AD,81,C6,6D,85,C6,8D,85,C6,CA,D0,DC
2730 DATA 60,EA,EA,EC,7F,C4,F0,1E,38,ED,7D,C4,48,98,ED,7E,C4,10,1C,A8,68,49,FF
2740 DATA 18,69,01,48,98,49,FF,69,00,AE,7F,C4,4C,EC,C6,18,6D,7D,C4,48,98
2750 DATA 6D,7E,C4,A8,68,60,EA,8D,9F,C1,AC,79,C4,AE,7A,C4,4C,07,C7,AD,77,C4
2760 DATA F0,04,88,88,E8,E8,C8,CA,AD,9E,C1,38,E9,19,8D,9E,C1,AD,9F,C1,E9,00
2770 DATA 8D,9F,C1,B0,E2,8C,79,C4,8E,7A,C4,60,A9,00,8D,79,C4,8D,77,C4,8D,7B,C4
2780 DATA 8D,7C,C4,8D,9E,C1,A9,FF,8D,7A,C4,8D,78,C4,AD,CD,C0,C9,65,90,01,60
2790 DATA C9,19,B0,06,20,F0,C6,4C,97,C7,38,E9,19,CE,79,C4,CE,77,C4,EE,7A,C4
2800 DATA EE,78,C4,CE,7C,C4,C9,19,B0,06,20,F0,C6,4C,97,C7,E9,19,EE,79,C4
2810 DATA EE,77,C4,CE,7B,C4,CE,7A,C4,CE,78,C4,C9,19,B0,06,20,F0,C6,4C,97,C7
2820 DATA E9,19,CE,79,C4,CE,77,C4,EE,7A,C4,EE,78,C4,EE,7C,C4,20,F0,C6,AD,CE,C0
2830 DATA 38,ED,CD,C0,B0,02,69,64,8D,9F,C1,A9,00,8D,9E,C1,AD,CB,C0,CD,CC,C0
2840 DATA B0,03,AD,CC,C0,A2,00,C9,00,F0,0E,8D,76,C4,A9,A3,38,ED,76,C4,90,03
2850 DATA E8,B0,F8,E0,00,D0,01,E8,8E,82,C6,AC,CF,C0,D0,01,C8,8C,80,C6,A9,00
2860 DATA 8D,83,C6,8D,81,C6,20,90,C6,AD,82,C6,8D,CF,C0,8D,76,C4,AD,CC,C0,D0,06
2870 DATA AD,CB,C0,8D,CC,C0,AC,7A,C4,B9,80,C4,8D,80,C6,B9,80,C5,8D,81,C6
2880 DATA AD,CB,C0,8D,82,C6,A9,00,8D,83,C6,20,90,C6,AD,84,C6,8D,7D,C4,A9,00
2890 DATA 8D,7E,C4,AA,AD,7C,C4,8D,7F,C4,AD,C8,C0,AC,C9,C0,20,C0,C6,E0,00,F0,03
2900 DATA 4C,85,C8,8D,AE,C1,8C,AF,C1,AC,79,C4,B9,80,C4,8D,80,C6,B9,80,C5
2910 DATA 8D,81,C6,AD,CC,C0,8D,82,C6,A9,00,8D,83,C6,20,90,C6,AD,84,C6,8D,7D,C4
2920 DATA A9,00,8D,7E,C4,A8,AA,AD,7B,C4,8D,7F,C4,AD,CA,C0,20,C0,C6,E0,00,F0,03
2930 DATA 4C,85,C8,C0,01,B0,06,8D,AD,C1,20,2F,C2,A2,01,BD,77,C4,D0,16,BD,79,C4
2940 DATA C9,FF,D0,09,DE,77,C4,DE,79,C4,4C,BF,C8,FE,79,C4,4C,BF,C8,BD,79,C4
2950 DATA D0,15,FE,77,C4,FE,79,C4,BC,7B,C4,F0,02,C8,C8,88,98,9D,7B,C4,4C,BF,C8
2960 DATA DE,79,C4,CA,F0,C5,A9,19,8D,7D,C4,A2,FF,8E,7F,C4,E8,8E,7E,C4,AD,9E,C1
2970 DATA AC,9F,C1,20,C0,C6,E0,00,F0,11,A9,00,8D,CD,C0,8D,CF,C0,8D,CC,C0,A9,64
2980 DATA 8D,CE,C0,60,8D,9E,C1,8C,9F,C1,CE,76,C4,D0,8C,AD,CF,C0,8D,76,C4
2990 DATA 4C,F9,C7,*
```

DATAMAKER

As a person who writes many programs in machine code for the Commodore 64 I have found a great need for a utility program that will convert a block of data into Basic DATA statements. The attached program does this very quickly.

It is written entirely in machine language and doesn't interfere with the Commodore development kit, on which it was developed.

To use this utility type in the BASIC loader and save it. Run it and after a while, if all goes well, the usual READY prompt will appear. The DATAMAKER utility is now installed and is ready to use at any time until the computer is turned off.

To call the routine type:

SYS 50768, first, last, line number where 'first' is the decimal value of the starting location of the block of data to be converted into DATA statements. 'Last' is the decimal value of the last element in the block of data to be converted and 'line number' is the line number that you would like the DATA statements to start numbering from.

There is only one restriction that I am aware of with this routine that the user should know about. The three parameters, apart from being in decimal, must be less than 32767 or an illegal quantity error is issued. This might pose problems if your data is located in the top half of the computer's memory or you want very large line numbers.

The problem is caused by one of the BASIC ROM routines used to process the parameters. To overcome this you must play a little trick and enter the desired 'large' number in a modified form. Let's say that you want the DATA statements to start numbering from 50000. Instead of entering 50000 as the start value you must calculate 50000 – 65536, which is –15536, and use this as the value. You might enter, for the previous example,

SYS 50768, 832, 895, –15536

So, if the number to be used is greater than 32767, then you must subtract 65536 from it and use that value.

If you want you can get the computer to work the value out for you. For example, this would achieve the same result as before

SYS 50768, 832, 895, 50000-65536

That's all there is to it!

Peter Thacker
Burchip VIC

```
10 PRINT"■■■           DATAMAKER"
20 PRINT"■          P.THACKER (1983)"
30 PRINT"■■TO CALL,TYPE"
40 PRINT"■     SYS 50768, FIRST,LAST,LINE NOS"
50 PRINT"■WHERE FIRST= START LOCATION TO CONVERT"
60 PRINT"      LAST= LAST LOCATION TO CONVERT"
70 PRINT"      LINE NOS= FIRST LINE NUMBER IN THE"
80 PRINT"■                 RESULTANT BASIC PROGRAM"
90 PRINT"■BASIC LINES GO IN STEPS OF 10"
100 PRINT"■THIS ROUTINE DOESNT INTERFERE WITH"
110 PRINT"COMMODORE'S DEVELOPMENT KIT"
120 PRINT"■PLEASE WAIT WHILE IT IS LOADED"
130 GOSUB150
140 NEW
150 REM DATA LOADER
160 LOC=50768
170 READX:IFX<0THEN230
180 CS=CS+X
190 PRINT"■CHECK SUM= ";CS;" HEADING FOR ";71810
200 POKELOC,X
210 LOC=LOC+1
220 GOTO170
230 IFCS=71810THEN 280
240 PRINT"■CHECKSUM ERROR..."
250 PRINT"■CHECK DATA STATEMENTS IN LINES"
260 PRINT"■20000-29999"
270 END
280 LOC=53081:CS=0:PRINT"■
290 READX:IFX<0THEN350
300 CS=CS+X
310 PRINT"■CHECK SUM= ";CS;" HEADING FOR ";16389;"
320 POKELOC,X
330 LOC=LOC+1
340 GOTO290
350 IFCS=16389THEN 400
360 PRINT"■CHECKSUM ERROR..."
370 PRINT"■CHECK DATA STATEMENTS IN LINES"
380 PRINT"■30000-39999"
390 END
400 RETURN
20000 DATA 32,228,199,165,101,141,250
20010 DATA207,165,100,141,251,207, 32
20020 DATA228,199,165,101,141,248,207
20030 DATA165,100,141,249,207, 32,228
20040 DATA199,165,101,141,254,207,165
20050 DATA100,141,255,207,165, 43,141
20060 DATA252,207,165, 44,141,253,207
20070 DATA173,252,207,141,246,207,173
20080 DATA253,207,141,247,207, 32,238
20090 DATA199, 32,238,199,173,252,207
20100 DATA133,251,173,253,207,133,252
20110 DATA160,  0,173,254,207,145,251
20120 DATA173,255,207,200,145,251, 24
20130 DATA173,254,207,105, 10,141,254
20140 DATA207,173,255,207,105,  0,141
20150 DATA255,207, 32,238,199, 32,238
20160 DATA199,173,252,207,133,251,173
20170 DATA253,207,133,252,160,  0,169
20180 DATA131,145,251, 32,238,199,162
20190 DATA  7,160,  0,173,252,207,133
20200 DATA251,173,253,207,133,252,173
20210 DATA250,207,133,253,173,251,207
20220 DATA133,254,177,253,141,170,  2
20230 DATA 32, 89,207,238,250,207,208
20240 DATA  3,238,251,207, 56,173,248
20250 DATA207,237,250,207,173,249,207
20260 DATA237,251,207,144,101, 24,173
20270 DATA252,207,105,  3,141,252,207
20280 DATA173,253,207,105,  0,141,253
20290 DATA207,173,252,207,133,251,173
20300 DATA253,207,133,252,160,  0,169
20310 DATA 44,145,251, 32,238,199,202
20320 DATA208,164, 56,173,252,207,233
20330 DATA  1,141,252,207,173,253,207
20340 DATA233,  0,141,253,207,173,252
20350 DATA207,133,251,173,253,207,133
20360 DATA252,160,  0,169,  0,145,251
20370 DATA 32,238,199,173,246,207,133
20380 DATA253,173,247,207,133,254,160
20390 DATA  0,173,252,207,145,253,200
20400 DATA173,253,207,145,253, 76,129
20410 DATA198, 24,173,252,207,105,  3
20420 DATA141,252,207,173,253,207,105
20430 DATA  0,141,253,207,173,252,207
20440 DATA133,251,173,253,207,133,252
20450 DATA160,  0,152,145,251,200,145
20460 DATA251,200,145,251, 32,238,199
20470 DATA173,246,207,133,253,173,247
20480 DATA207,133,254,160,  0,173,252
20490 DATA207,145,253,200,173,253,207
20500 DATA145,253, 24,173,252,207,105
20510 DATA  2,141,252,207,173,253,207
20520 DATA105,  0,141,253,207,173,252
20530 DATA207,133, 45,173,253,207,133
20540 DATA 46,169, 67,141,119,  2,169
20550 DATA 76,141,120,  2,169, 82,141
20560 DATA121,  2,169, 13,141,122,  2
20570 DATA169,  4,133,198, 96, 32,115
20580 DATA  0, 32,158,173, 32,170,177
20590 DATA 96,238,252,207,208,  3,238
20600 DATA253,207, 96
20610 DATA-1
30000 DATA173,167,  2, 72,173,158,  2
30010 DATA 72,173,169,  2, 72,169, 32
30020 DATA141,167,  2,141,168,  2,169
30030 DATA 48,141,169,  2,173,170,  2
30040 DATA201,  0,240, 72,238,169,  2
30050 DATA173,169,  2,201, 58,240,  6
30060 DATA206,170,  2, 76,114,207,169
30070 DATA 48,141,169,  2,173,168,  2
30080 DATA201, 32,208,  8,169, 49,141
30090 DATA168,  2, 76,131,207,238,168
30100 DATA  2,173,168,  2,201, 58,208
30110 DATA220,169, 48,141,168,  2,173
30120 DATA167,  2,201, 32,208,  8,169
30130 DATA 49,141,167,  2, 76,131,207
30140 DATA238,167,  2, 76,131,207,160
30150 DATA  0,173,167,  2,145,251,173
30160 DATA168,  2,200,145,251,173,169
30170 DATA  2,200,145,251,104,141,169
30180 DATA  2,104,141,168,  2,104,141
30190 DATA167,  2, 96
30200 DATA-1
```

TDIR64

Got a tape with files on it but you don't know what files or where? Or perhaps you just want a neat printed listing of the files on the tape for your records.

TDIR64 (for Tape Directory Program for the Commodore 64) provides a catalogue of files on a tape. The output includes a user supplied identifier for the tape, the date, file names, type (program or data), the file size (in bytes) and optionally the tape counter at the start of the file.

The program asks you for the output device, Y = Printer, N = screen. Answer Y to the LOG TAPE COUNTER prompt to select that option. Up to 16 characters may be entered for the tape identifier. The date must be in an 8 character format.

The files open at line 300 opens the next file on tape. The name of the file is found starting at byte 5 in the tape buffer (lines 320 – 350). Bytes are numbered starting at 0. By PEEKing the tape buffer I found that programs are flagged by a 1 in byte 0 (line 510). Bytes 1 and 2 contain the program start address in low byte, high byte order (line 580). Bytes 3 and 4 contain the start of BASIC variables address (line 585). The difference is the program size (line 590). The size of data files is calculated by counting bytes (lines 540 – 560).

If the tape counter option is requested the program prompts the user to enter the counter. This is adjusted to allow for the first block containing the file name (lines 420 – 480).

The program loops (line 660) until the RUN/STOP key is pressed.

Lines 1000 – 8999 contain I/O subroutines.

Richard Wooler
Paraburdoo WA

```
10 REM ********************************
20 REM *                              *
30 REM *        T D I R 6 4           *
40 REM *                              *
50 REM *   TAPE DIRECTORY LISTING     *
60 REM *                              *
70 REM *   (C) RICHARD WOOLLER        *
80 REM *                              *
85 REM *   PARABURDOO   14 SEP 1983   *
86 REM *                              *
90 REM ********************************
100 TP=1:SC=3:PR=4:REM >> DEVICES
110 IN=1:OU=2:      REM >> LOGICAL UNITS
120 RD=0:UG=1:      REM >> ACCESS MODES
130 TB=828:         REM >> TAPE BUFFER
190 GOSUB 2000:PRINT "■■"+L$+"■"
200 REM >> PROMPT FOR OUTPUT DEVICE
210 PRINT "PRINTER OUTPUT (Y/N)?"
220 GOSUB 1000:GOSUB 8000:LP%=YES%
230 OD=SC:IF LP% THEN OD=PR
240 PRINT "LOG TAPE COUNTER (Y/N)?"
250 GOSUB 1000:GOSUB 8000:TC%=YES%
280 OPEN OU,OD,UG:GOSUB 2000
290 GOSUB 3000
292 REM >> OPEN FIRST FILE AND DELETE
294 REM >> PRESS PLAY ON TAPE PROMPT.
296 OPEN IN,TP,RD: REM >> OPEN FILE
298 GOSUB 8000:GOSUB 8000:GOTO 320
300 OPEN IN,TP,RD: REM >> OPEN FILE
310 REM >> GET FILE NAME FROM BUFFER
320 F$=""
330 FOR A=TB+5 TO TB+20
340 F$=F$+CHR$(PEEK(A))
350 NEXT A
400 REM >> GET TAPE COUNTER
410 TS$="###"
420 IF NOT TC% THEN 500
430 INPUT "TAPE COUNTER";CT:GOSUB 8000
440 IF CT<5 OR CT>999 THEN 430
445 REM >> CORRECT FOR FIRST RECORD
450 CT=CT-5:
460 TS$=STR$(CT)
470 TS$=RIGHT$(TS$,LEN(TS$)-1)
480 TS$=RIGHT$("00"+TS$,3)
500 REM >> GET FILE SIZE
510 PR%=PEEK(TB)=1:REM >> PROGRAM?
520 IF PR% THEN 570
530 REM >> DATA FILE, COUNT BYTES
540 BY=0:TY$="DATA    "
550 GET#IN,C$:BY=BY+1:IF ST=0 THEN 550
560 GO TO 600
570 REM >> PROGRAM FILE SIZE IN HEADER
575 TY$="PROGRAM "
580 PS=PEEK(TB+1)+PEEK(TB+2)*256
585 PE=PEEK(TB+3)+PEEK(TB+4)*256
590 BY=PE-PS
600 REM >> CLOSE FILE AND PRINT LINE
610 CLOSE IN
620 B$=RIGHT$("    "+STR$(BY),5)
630 L$=TS$+"  "+F$+"  "+TY$+"  "+B$
640 GOSUB 5000
660 GOTO 300
1000 REM ** SUBROUTINE YES/NO ANSWER **
1010 REM >> ANSWER RETURNED IN YES%
1020 GET K$:IF K$="" THEN 1020
1030 YES%=K$="Y":IF YES% THEN 1050
1040 IF K$<>"N" THEN 1010
1050 GET K$:IF K$<>"" THEN 1050
1999 RETURN
2000 REM ** SUBROUTINE HEADER **
2020 L$="** TDIR64 - TAPE "
2030 L$=L$+"DIRECTORY LISTING - **"
2999 RETURN
3000 REM ** SUBROUTINE TAPE NAME **
3002 IF NOT LP% THEN 3010
3004 GOSUB 2000:GOSUB 5000
3006 L$="":GOSUB 5000
3010 INPUT "TAPE IDENTIFIER";TN$
3015 GOSUB 8000
3020 IF LEN(TN$)>=16 THEN 3040
3030 TN$=TN$+" ":GOTO 3020
3040 TN$=LEFT$(TN$,16)
3050 INPUT "DATE ";D$
3055 GOSUB 8000
3060 L$="TAPE : "+TN$
3070 L$=L$+"   DATE "+LEFT$(D$,8)
3090 GOSUB 5000:L$="":GOSUB 5000
3100 L$="LOC  FILE-NAME          "
3110 L$=L$+"TYPE       SIZE"
3120 GOSUB 5000
3130 L$="":GOSUB 5000
3999 RETURN
5000 REM ** SUBROUTINE PRINT LINE **
5010 M$="":IF LP% THEN M$="    "
5020 PRINT#OU,M$+L$
5999 RETURN
8000 REM ** SUBROUTINE DELETE LINE **
8010 L$="                "
8020 PRINT "■"+L$+L$+"■"
8999 RETURN
```

```
** TDIR64 - TAPE DIRECTORY LISTING - **

TAPE : TAPE#001              DATE 14/9/83

LOC  FILE-NAME          TYPE       SIZE

003  BREAKOUT 64        PROGRAM    7891
067  TDIR64             PROGRAM    2531
089  TDIR64 DOCUMENT    DATA       1879
```

CHARACTER MAKER

After hours of using a pencil and paper to make up my user defined graphics I decided to make up a program to do the job for me.

Using the program is easy – use the numerical keys to fill in a pixel as a co-ordinate and use the left arrow key and two co-ordinates to delete a pixel.

You may create your character by pressing F1 or use other features such as rolling the spike in four different directions or reversing the character and then creating the character.

Jarrad Webb
Henley Beach SA

```
5 PRINT"■"
9 CH=160:X=1104
10 PRINT"          R CHARACTER MAKER ■"
11 PRINT"■TYPE IN TWO NUMBERS TO SELECT THE SPACE■ TO BE FILLED IN."
12 PRINT"■PRESS ■R←■ AND THE CORDS. TO RUB OUT"
14 PRINT"■■■■■■■■■■R   PRESS SPACE TO START   ■"
15 GETA$:IFA$<>" "THEN15
19 PRINT"■Q■12345678"
20 FORI=1TO8
30 PRINT"TTTTTTTT"I:NEXT
32 PRINT"■■■■■TYPE IN ■F1■ TO ENTER THE CHARACTER"
33 PRINT"■■■■■TYPE IN ■F3■ TO REVERSE THE CHARACTER"
34 GOSUB650
35 GETA$:IFA$=""THEN35
36 GOSUB1000
37 IFA$="←"THENCH=80:GOTO35
38 IFA$="■"THEN200
39 IFA$="■"THENZ=1:GOSUB700
40 GETB$:IFB$=""THEN40
42 A=VAL(A$)
43 B=VAL(B$)
44 IFA=.ORB=.THEN35
46 A=A-1:B=B-1
60 Q=B*40
70 Q=1104+A+Q
80 POKEQ,CH
100 CH=160:GOTO35
200 FORJ=1103TO1383STEP40
210 FORI=1TO8
220 IFPEEK(J+I)<>160THENPOKEJ+I,32:GOTO300
230 READA,B
240 IFA=ITHENT=T+B:RESTORE:GOTO300
250 GOTO230
300 NEXTI
310 C=C+1:T(C)=T:T=0:NEXTJ
400 DATA8,1,7,2,6,4,5,8,4,16,3,32,2,64,1,128
530 PRINT"■■■■■■■■PLEASE WAIT !":GOTO550
550 REM
560 POKE52,48:POKE56,16
570 POKE56334,PEEK(56334)AND251
572 POKE1,PEEK(1)AND251
574 FORI=0TO511:POKEI+12288,PEEK(I+53248):NEXT
576 POKE1,PEEK(1)OR4
578 POKE56334,PEEK(56334)OR1
579 POKE53272,(PEEK(53272)AND240)+12
580 POKE53272,28
590 FORC=12288TO12288+7:U=U+1:POKEC,T(U):NEXT
600 PRINT"■Q■THE CHARACTER FOR THE DATA:"
610 FORI=1TO8
620 PRINT"■■■"T(I):NEXT
630 PRINT"■■IS THIS: @ @ @ @"
640 PRINT"■■■DO YOU WANT TO DO ANOTHER ONE (Y/N)"
650 GETA$:IFA$=""THEN650
660 IFA$="Y"THENPOKE53272,21:RUN
670 IFA$="N"THENPOKE53272,21:END
680 GOTO650
700 FORJ=1103TO1383STEP40
710 FORI=1TO8
720 IFPEEK(J+I)<>80THENPOKEJ+I,80:GOTO740
730 IFPEEK(J+I)<>160THENPOKEJ+I,160:GOTO740
740 NEXTI
745 NEXTJ:RETURN
750 FORJ=1103TO1383STEP40
755 FORI=1TO8
760 IFPEEK(J+I)<>80THENPOKE(J+I)+P,160:POKEJ+I,80
763 PRINT"■Q■12345678"
764 PRINT"■■■■■■■■■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■"
765 NEXTI:NEXTJ:RETURN
800 FORJ=1383TO1103STEP-40
810 FORI=8TO1STEP-1
815 PRINT"■■■■■■■■■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■"
820 IFPEEK(J+I)<>80THENPOKE(J+I)+P,160:POKEJ+I,80
830 NEXTI:NEXTJ:RETURN
850 PRINT"■■■■"TAB(15)"R W ■ = UP"
855 PRINT"■■■■■"TAB(15)"R Z ■ = DOWN"
860 PRINT"■■■■■■"TAB(15)"R A ■ = LEFT"
865 PRINT"■■■■■■■"TAB(15)"R S ■ = RIGHT"
870 RETURN
1000 IFA$="N"THENP=.40:GOSUB750
1010 IFA$="C"THENP=40:GOSUB800
1020 IFA$="A"THENP=1:GOSUB750
1030 IFA$="S"THENP=1:GOSUB800
1040 RETURN
```

TINYGRAPH

In developing programs on the Commodore 64 there has been a need to use built in, and easy to use, graphics commands. Below is a short program that inserts itself into the BASIC interpreter and adds these commands. To make the program short and to provide one-key commands three unused keys on the keyboard have been used rather than tokenising three, more meaningful, graphics words.

The commands are: & border colour, screen colour – which will clear the screen and replace it with a high resolution screen ready for drawing; 'x1,y1,x2,y2,colour – will draw a line between (x1,y1) and (x2,y2) in the specified colour; ! border colour, screen, character – will replace the high-res screen by the normal text screen and the characters will be printed in the specified colour.

Each colour is a value from 0 to 15 and corresponds to colours that can be produced by the C64. For example 0 = black, 1 = white, 2 = red and so on (i.e. one less than the number on the number keys). Depending upon the screen colour, the line may not appear the colour that you wanted.

The co-ordinates that are allowed for the x and y values are 0 to 319 for x and 0 to 199 for y with (0,0) being in the top left hand corner – so essentially x measures how far horizontally you've gone and y how far vertically you've travelled. Should illegal values be used then the draw command will automatically return to the normal text screen and display an ILLEGAL QUANTITY ERROR message.

A word of warning. If some other part of your program has a fault that causes an error message to be printed when you are in the graphics mode, the screen will not revert back to the text mode and a line of coloured squares will appear on the high-res screen in place of the message. You will have to manually restore the text screen by using the ! command.

All the values can be any legal BASIC expression and are not confined to simple numbers - except if the commands are used directly and not in a program (here they must be pure numbers). So, for example, in a program the colour code could be SQR(LOG (PEEK(24*1024)+1) but directly it must be 1 or 2 etc.

One other restriction is that, when used in programs, each of the commands must be preceded by a colon(:).

Program three shows examples of the use of these commands to draw lines, circles and to fill in regions.

To add these commands to BASIC type in the loader program (program one) exactly as

```
10 LOC=49152:PRINT"■■■■LOADING"
20 READD$:IFD$="*"THEN130
25 PRINT"■AT ";LOC;"HEADING FOR 50587"
30 V=0
40 FORI=1TO2
50 P$=MID$(D$,I,1)
60 GOSUB500
70 V=V*16+P
80 NEXT
90 POKELOC,V
100 T=T+V
110 LOC=LOC+1
120 GOTO20
130 IFT=164041THENSYS50475
140 PRINT"OOPS!...CHECK THE DATA"
150 END
500 P=(48-ASC(P$))*(P$<="9")
510 P=P+(55-ASC(P$))*(P$>="A")
520 RETURN
1000 DATAA9,4C,85,84,A9,28,85,85,A9
1010 DATAC0,85,86,60,26,53,45,5C,45
1020 DATA58,54,5F,52,41,57,E6,7A,D0
1030 DATA02,E6,7B,60,20,9E,AD,20,AA
1040 DATAB1,A5,65,60,EA,C9,26,D0,03
1050 DATA4C,6D,C0,C9,21,D0,03,4C,D0
1060 DATAC0,C9,27,D0,03,4C,96,C1,EA
1070 DATA38,E9,30,4C,87,00,C0,F4,4C
1080 DATAD0,C0,A0,00,B1,7A,D9,14,C0
1090 DATAD0,08,C8,C0,01,D0,F4,4C,96
1100 DATAC1,C9,3A,B0,0A,C9,20,F0,07
1110 DATA38,E9,30,38,E9,D0,60,4C,73
1120 DATA00,20,18,C0,4C,79,C0,20,18
1130 DATAC0,20,18,C0,20,1F,C0,8D,20
1140 DATAD0,20,18,C0,20,1F,C0,85,FD
1150 DATAAD,11,D0,09,20,8D,11,D0,AD
1160 DATA18,D0,29,F0,09,08,8D,18,D0
1170 DATAAD,00,DD,29,FC,09,03,8D,00
1180 DATADD,A9,20,85,FC,AA,A9,00,85
1190 DATAFB,A0,00,91,FB,88,D0,FB,E6
1200 DATAFC,CA,D0,F6,A9,04,AA,85,FC
1210 DATAA0,00,A5,FD,91,FB,88,D0,FB
1220 DATAE6,FC,CA,D0,F6,A9,3A,4C,79
1230 DATA00,20,18,C0,4C,DC,C0,20,18
1240 DATAC0,20,18,C0,20,1F,C0,8D,20
1250 DATAD0,20,18,C0,20,1F,C0,8D,21
1260 DATAD0,20,18,C0,20,1F,C0,8D,86
1270 DATA02,A9,93,20,D2,FF,AD,11,D0
1280 DATA29,DF,8D,11,D0,AD,00,DD,29
1290 DATAFC,09,03,8D,00,DD,AD,18,D0
1300 DATA29,F0,09,04,8D,18,D0,4C,CB
1310 DATAC0,20,04,C5,A5,66,D0,18,20
1320 DATA0F,BC,A5,FC,A4,FB,20,91,B3
1330 DATA20,5C,C1,20,0E,C5,A5,66,F0
1340 DATA04,20,51,C1,60,A9,00,A0,01
1350 DATA20,91,B3,A9,FF,85,66,20,4C
1360 DATAC4,60,A2,06,B5,60,9D,41,03
1370 DATACA,D0,F8,60,A2,06,BD,41,03
1380 DATA95,60,CA,D0,F8,60,A5,66,45
1390 DATA6E,85,6F,A5,61,60,AD,40,03
1400 DATA85,FB,AD,3D,03,8D,40,03,A5
1410 DATAFB,8D,3D,03,AD,3F,03,85,FB
1420 DATAAD,3C,03,8D,3F,03,A5,FB,8D
1430 DATA3C,03,AD,41,03,85,FB,AD,3E
1440 DATA03,8D,41,03,A5,FB,8D,3E,03
1450 DATA60,20,18,C0,4C,A2,C1,20,18
1460 DATAC0,20,18,C0,20,9E,AD,A9,01
1470 DATA85,FC,A9,40,85,FB,20,18,C1
1480 DATA20,AA,B1,A5,65,8D,3C,03,A5
1490 DATA64,8D,3D,03,20,18,C0,20,9E
1500 DATAAD,A9,00,85,FC,A9,C8,85,FB
1510 DATA20,18,C1,20,AA,B1,A5,65,8D
1520 DATA3E,03,20,18,C0,20,9E,AD,A9
1530 DATA01,85,FC,A9,40,85,FB,20,18
1540 DATAC1,20,AA,B1,A5,65,8D,3F,03
1550 DATAA5,64,8D,40,03,20,18,C0,20
1560 DATA9E,AD,A9,00,85,FC,A9,C8,85
1570 DATAFB,20,18,C1,20,AA,B1,A5,65
1580 DATA8D,41,03,20,18,C0,20,1F,C0
1590 DATA0A,0A,0A,0A,8D,48,03,AD,3C
1600 DATA03,CD,3F,03,D0,15,AD,3D,03
1610 DATACD,40,03,D0,0D,A2,06,A9,00
1620 DATA9D,41,03,CA,D0,FA,4C,A6,C2
1630 DATAAD,40,03,AC,3F,03,20,91,B3
1640 DATA20,0F,BC,AD,3D,03,AC,3C,03
1650 DATA20,91,B3,20,5C,C1,20,53,B8
1660 DATA20,46,C1,A9,00,AC,41,03,20
1670 DATA91,B3,20,0F,BC,A9,00,AC,3E
1680 DATA03,20,91,B3,20,5C,C1,20,53
1690 DATAB8,20,0F,BC,20,51,C1,20,5C
1700 DATAC1,20,12,BB,20,46,C1,A9,00
1710 DATA85,66,EA,EA,20,AA,B1,A9,00
1720 DATAC5,65,D0,07,C5,64,D0,03,4C
1730 DATAE7,C2,A9,00,A0,01,20,91,B3
1740 DATA20,71,C4,20,5C,C1,20,12,BB
1750 DATA20,46,C1,4C,E3,C4,CD,3E,03
1760 DATAB0,03,20,65,C1,AD,41,03,CD
1770 DATA3E,03,B0,03,4C,CB,C0,20,35
1780 DATAC3,4C,C0,C4,AC,3C,03,20,91
1790 DATAB3,20,0F,BC,20,51,C1,20,5C
1800 DATAC1,20,6A,B8,20,AA,B1,A5,65
1810 DATA8D,3C,03,A5,64,8D,3D,03,EE
1820 DATA3E,03,4C,B1,C2,20,9B,C4,4C
1830 DATAF9,C2,EA,EA,EA,EA,EA,EA,EA
1840 DATAEA,EA,EA,EA,38,AD,3F,03
1850 DATAED,3C,03,AD,40,03,ED,3D,03
1860 DATAB0,03,4C,CB,C0,20,35,C3,4C
1870 DATA78,C4,3E,03,20,91,B3,20,0F
1880 DATABC,20,51,C1,20,5C,C1,20,6A
1890 DATAB8,20,AA,B1,A5,65,8D,3E,03
1900 DATAEE,3C,03,D0,03,EE,3D,03,4C
1910 DATAF9,C2,18,AD,3D,03,6A,AD,3C
1920 DATA03,6A,4A,4A,8D,49,03,AD,3E
1930 DATA03,4A,4A,4A,8D,4A,03,A9,00
1940 DATAAC,49,03,20,91,B3,20,0F,BC
1950 DATAA9,00,A0,08,20,91,B3,20,5C
1960 DATAC1,20,30,BA,20,AA,B1,A5,65
1970 DATA8D,4B,03,A5,64,8D,4C,03,A9
1980 DATA00,AC,4A,03,20,91,B3,20,0F
```

it is written and save it on tape or disk under the name LOADER. Type RUN and press return. The loader contains a checksum routine to prevent it from working if an error has been made – if this has happened then check each data statement again, save it and try again.

If everything goes to plan then the screen will clear and the name: 'TINYGRAPH BY P.THACKER' will appear. You now have these three commands at your disposal. The LOADER takes a long time to POKE the instructions into the area of RAM located above the BASIC interpreter, at 49152, but there is a way to make it load almost instantaneously.

Here's what to do: RUN the LOADER (or follow the above instructions) and verify that it works; Type SYS 64738 and press return. This restores the computer to its just turned on condition without losing the graphics; Type POKE44,100:POKE 25600,0:NEW and press return; Type in program two and save it on tape or disk; RUN it and if it's OK then the number 6471 will appear – if it doesn't then check the DATA statements again; Type SYS 828 and press return.

Finally, make these changes to program two and RUN it again.

```
10 LOC = 2068
60 DATA 253,145,251
90 DATA 76,43,197
```

and a value of 6961 should appear. If not, check it. If all has gone to plan so far, type POKE 44,0:NEW and enter this one line program 10 SYS 2068. Now, to finish off, type POKE 45,160:POKE 46,14:CLR and press return. Now SAVE this program on disk or tape, under "TINYGRAPH".

What you have done is copied the graphics routines into the area used normally for your BASIC programs and joined onto it a small machine code program that will correctly reposition it. To use it just load TINYGRAPH as you would a normal BASIC program and type RUN and within a second you'll be off and running.

A brief comment about how the program works is important. The BASIC interpreter uses a subroutine located in zero page called the CHRGET routine to get the next character in any BASIC program. This routine is altered so that when one of the new commands is found the program you've just entered carries out the instructions required. When you load and run TINYGRAPH the bottom of BASIC is moved up from location 2048 to 16384 to make room for the 8192 bytes needed for the screen. If you're quick you'll notice that some 6K is unused (2048 to 8191) below the screen. This was done to avoid the character generator appearing on the screen.

Peter Thacker
Birchip VIC

```
1990 DATABC,A9,01,A0,40,20,91,B3,20
2000 DATA5C,C1,20,30,BA,20,0F,BC,AD
2010 DATA4C,03,AC,4B,03,20,91,B3,20
2020 DATA5C,C1,20,6A,B8,20,0F,BC,A9
2030 DATA20,A0,00,20,91,B3,20,5C,C1
2040 DATA20,6A,B8,20,0F,BC,A9,07,2D
2050 DATA3E,03,A8,A9,00,20,91,B3,20
2060 DATA5C,C1,20,6A,B8,20,AA,B1,A5
2070 DATA65,8D,4B,03,A5,64,8D,4C,03
2080 DATAAD,3C,03,29,07,85,FD,38,A9
2090 DATA07,E5,FD,85,FD,A9,00,85,FE
2100 DATA38,26,FE,A5,FD,F0,07,A4,FD
2110 DATA06,FE,88,D0,FB,A0,00,AD,4B
2120 DATA03,85,FB,AD,4C,03,85,FC,B1
2130 DATAFB,05,FE,91,FB,A9,00,AC,4A
2140 DATA03,20,91,B3,20,0F,BC,A9,00
2150 DATAA0,28,20,91,B3,20,5C,C1,20
2160 DATA30,BA,20,0F,BC,A9,00,AC,49
2170 DATA03,20,91,B3,20,5C,C1,20,6A
2180 DATAB8,20,0F,BC,A9,04,A0,00,20
2190 DATA91,B3,20,5C,C1,20,6A,B8,20
2200 DATAAA,B1,A5,65,85,FB,A5,64,85
2210 DATAFC,A0,00,AD,48,03,11,FB,91
2220 DATAFB,60,AD,00,DD,29,FC,09,03
2230 DATA8D,00,DD,AD,11,D0,29,DF,8D
2240 DATA11,D0,PD,18,D0,29,F0,09,04
2250 DATA8D,18,D0,A9,93,20,D2,FF,20
2260 DATA71,BF,60,20,0F,BC,20,51,C1
2270 DATA60,A2,06,BD,56,03,95,60,CA
2280 DATAD0,F8,20,0F,BC,20,51,C1,20
2290 DATA5C,C1,20,6A,B8,A2,06,B5,60
2300 DATA9D,56,03,CA,D0,F8,4C,22,C3
2310 DATA38,AD,3F,03,ED,2C,03,AD,40
2320 DATA03,ED,3D,03,B0,03,20,65,C1
2330 DATAA9,00,AC,3E,03,20,91,B3,A2
2340 DATA06,B5,60,9D,56,03,CA,D0,F8
2350 DATA60,A2,06,BD,50,02,95,60,CA
2360 DATAD0,F8,20,0F,BC,20,51,C1,20
2370 DATA5C,C1,20,6A,B8,A2,06,B5,60
2380 DATA9D,50,03,CA,D0,F8,4C,D4,C2
2390 DATAAD,41,03,CD,3E,03,B0,03,20
2400 DATA65,C1,AD,2D,03,AC,3C,03,20
2410 DATA91,B3,A2,06,B5,60,9D,50,02
2420 DATACA,D0,F8,4C,B1,C2,20,46,C1
2430 DATAA5,66,29,80,85,66,60,20,53
2440 DATAB8,4C,07,C5,8D,52,45,4F,43
2450 DATA41,48,54,2E,50,20,59,42,20
2460 DATA48,50,41,52,47,59,4E,49,54
2470 DATAA9,00,8D,20,D0,A9,0B,8D,21
2480 DATAD0,A9,40,85,2C,A9,00,8D,00
2490 DATA40,A9,93,20,D2,FF,A2,0C,A9
2500 DATA11,20,D2,FF,CA,D0,FA,A2,08
2510 DATAA9,1D,20,D2,FF,CA,D0,FA,A9
2520 DATAC5,85,FC,A9,13,85,FB,A0,17
2530 DATAB1,FB,20,D2,FF,88,D0,F8,A9
2540 DATA08,85,FB,A2,00,A0,00,88,D0
2550 DATAFD,CA,D0,F8,C6,58,D0,F2,20
2560 DATA00,C0,A9,93,8D,77,02,A9,4E
2570 DATA8D,78,02,A9,45,8D,79,02,A9
2580 DATA57,8D,7A,02,A9,0D,8D,7B,02
2590 DATAA9,05,85,C6,60,*
```

```
10 LOC=828
20 READX:T=T+X:IFX<0THENPRINTT:END
30 POKELOC,X:LOC=LOC+1:GOTO20
40 DATA169,0,133,251,133,253,168,169
50 DATA192,133,252,169,9,133,254,177
60 DATA251,145,253
70 DATA230,251,230,253,208,4,230,252
80 DATA230,254,165,252,201,198,208,236
90 DATA96
100 DATA-1
```

```
10 :&0,0:REM BLACK BORDER,SCREEN
15 REM DRAW HOUSE
20 :50,50,150,50,8
30 :150,50,150,150,8
40 :150,150,50,150,8
50 :50,150,50,50,8
60 :50,50,100,25,8
70 :100,25,150,50,8
80 :60,150,60,100,8
90 :60,100,90,100,8
100 :90,100,90,150,8
105 REM FILL IN DOOR
110 FORY=100TO150
120 :60,Y,90,Y,3
130 NEXT
135 REM DRAW WINDOW
140 :100,120,100,80,5
150 :100,80,130,80,5
160 :130,80,130,120,5
170 :130,120,100,120,5
180 :100,100,130,100,5
190 :115,120,115,80,5
195 REM DRAW THE SUN
200 FORX=221TO279
210 F=SQR(ABS(900-(X-251)*(X-251)))
215 F1=SQR(ABS(900-(X-249)*(X-249)))
220 :X,50-F,X,50-F1,7
230 :X,50+F,X,50+F1,7
240 NEXT
250 REM WAIT FOR KEYPRESS
260 GETA$:IFA$=""THEN260
270 :10,11,14:REM TEXT BLACK,GREY,BLUE
280 LIST
```

SCREEN PLOTTER

When you first enter the program and run it garbage will appear on the screen. Then it will start to clear itself slightly.

Then the screen will turn cyan and one little pixel will appear in the upper left of the screen.

This pixel can be moved around by: *INSERT FIGURE 1.

If you wish to erase anything that you have drawn, press the E key, and your pixel will flash. If you now retrace any or all of your steps the pixels will be erased. This can also be used for moving your pixel without drawing.

To start drawing again press the D key.

If you wish to start again you can either press the HOME key or Run stop and Restore and rerun the program.

Jarrad Webb
Henley Beach

COMMODORE 64 SCREEN PLOTTER

```
10 R=1:B=2*4096:POKE53272,PEEK(53272)OR8:POKE53265,PEEK(53265)OR32:GOSUB70
20 FORI=1024TO2023:POKEI,3:NEXT:GOTO80
30 CH=INT(X/8):RO=INT(Y/8):LN=YAND7:BY=B+RO*320+8*CH+LN:BI=7-(XAND7)
40 IFR=1THENPOKEBY,PEEK(BY)OR(2↑BI)
50 IFR=.THENPOKEBY,PEEK(BY)OR(2↑BI):POKEBY,0
60 RETURN
70 FORI=BTOB+7999:POKEI,.:NEXT:RETURN
80 M=PEEK(197)
90 IFM=14THENR=.:GOSUB30
100 IFM=18THENR=1:GOSUB30
110 IFM=33THENY=Y-1:GOSUB30
120 IFM=30THENX=X-1:Y=Y-1:GOSUB30
130 IFM=36THENY=Y+1:GOSUB30
140 IFM=38THENX=X+1:Y=Y-1:GOSUB30
150 IFM=34THENX=X-1:GOSUB30
160 IFM=39THENX=X-1:Y=Y+1:GOSUB30
170 IFM=37THENX=X+1:GOSUB30
180 IFM=47THENX=X+1:Y=Y+1:GOSUB30
190 IFM=51THENGOSUB70
200 IFM=64THENGOSUB30
210 GOSUB230
220 GOTO80
230 IFX<1THENPOKEBY,.:X=1
240 IFX>318THENPOKEBY,.:X=X-2
250 IFY<1THENPOKEBY,.:Y=1
260 IFY>199THENPOKEBY,.:Y=Y-2
270 RETURN
```

DAYS TO GO

HEWLETT PACKARD

'How long to go till our holidays?', 'Only five weeks to Christmas!'. Days to go is a simple program – enter any two dates and it will list every day in between, giving you the days and weeks to go plus an optional column of percentages. The printout is an invaluable gift to someone getting married, expecting a baby or waiting for an important date to arrive. It occupies pride of place on many of my friends' toilet doors!

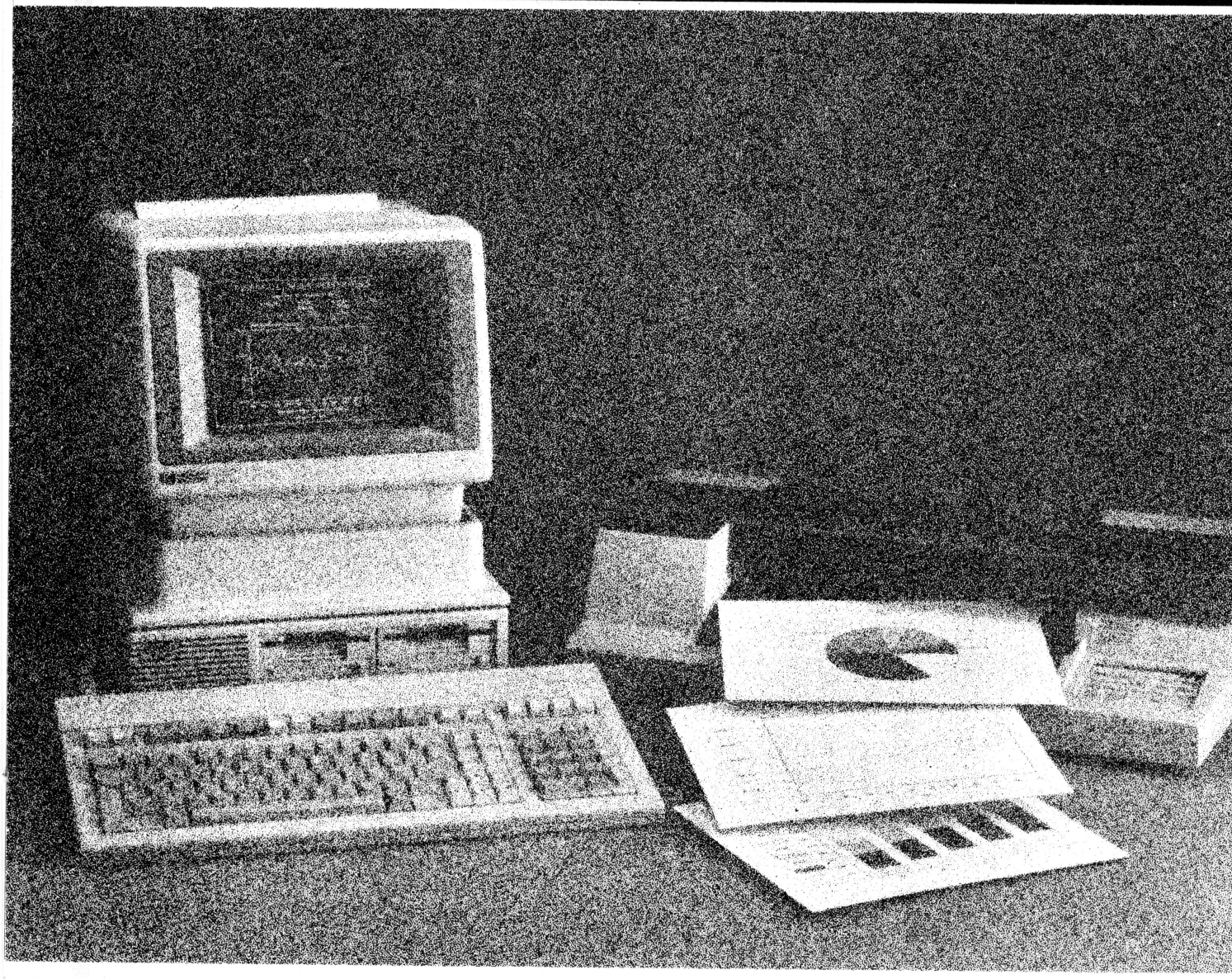
Written in Hewlett-Packard BASIC, there is little that would not work on most other BASICS-the DO/DOEND and IMAGE statements won't, but it should be obvious what they're doing.

The three date subroutines at lines 1000, 2000 and 3000 are not original. I've converted them from FORTRAN listings I've had for years. They are the sort of routine that most programmers need in their bag of tricks.

Phil Carter
Warrnambool VIC

```
DAYSTOGO
   10 REM*************************************************
   20 REM*                                               *
   30 REM* PROGRAM : DAYS TO GO                          *
   40 REM* AUTHOR  : PHIL CARTER                         *
   50 REM* DATE    : NOVEMBER 1983                       *
   60 REM*                                               *
   70 REM* THIS PROGRAM WILL PRINT ALL THE DAYS THAT OCCUR  *
   80 REM* BETWEEN ANY TWO DATES.                        *
   90 REM* THE NUMBER OF DAYS TO GO AND THE NUMBER OF WEEKS *
  100 REM* TO GO TO THE FINAL DAY IS PRINTED ALONGSIDE EACH *
  110 REM* DATE.                                        *
  120 REM*                                              *
  130 REM*************************************************
  140 DIM A1$[1],D1$[7,9],M1$[12,9]
  150 FOR I1=1 TO 7
  160   READ D1$[I1]
  170 NEXT I1
  180 FOR I1=1 TO 12
  190   READ M1$[I1]
  200 NEXT I1
  210 PRINT
  220 PRINT "**** DAYS TO GO ****"
  230 PRINT
  240 INPUT "Instructions? type YES or NO:",A1$
  250 IF A1$<>"Y" AND A1$<>"N" THEN 230
  260 IF A1$="N" THEN 330
  270 PRINT
  280 PRINT "'DAYS TO GO' is a computer program that will first ask"
  290 PRINT "you  to enter any two dates. Then, it will list all the"
  300 PRINT "days in between the two days, giving the number of days"
  310 PRINT "to go,  the number of weeks to go and,  if you wish,  a"
  320 PRINT "column of percentages."
  330 PRINT
  340 INPUT "Enter start date - DD,MM,YYYY:",D1,M1,Y1
  350 IF D1<1 OR D1>31 THEN 330
  360 IF M1<1 OR M1>12 THEN 330
  370 IF Y1<1900 THEN 330
  380 IF (M1=4 OR M1=6 OR M1=9 OR M1=11) AND D1=31 THEN 330
  390 IF M1=2 AND D1>29 THEN 330
  400 GOSUB 2000
  410 S1=A1
  420 PRINT
  430 INPUT "Enter finish date - DD,MM,YYYY:",D1,M1,Y1
  440 IF D1<1 OR D1>31 THEN 420
  450 IF M1<1 OR M1>12 THEN 420
  460 IF Y1<1900 THEN 420
  470 IF (M1=4 OR M1=6 OR M1=9 OR M1=11) AND D1=31 THEN 420
  480 IF M1=2 AND D1>29 THEN 420
  490 GOSUB 2000
  500 F1=A1
```

PROGRAMS FOR HEWLETT-PACKARD



```
510 IF F1>=S1 THEN 550
520 PRINT
530 PRINT "Can't have finish date earlier than start date"
540 GOTO 330
550 W1=INT((F1-S1)/7)+1
560 PRINT
570 INPUT "Print percentages?:",A1$
580 IF A1$<>"Y" AND A1$<>"N" THEN 560
590 PRINT
600 IF A1$="N" THEN DO
610   PRINT USING 680;"DATE","DAYS TO GO","WEEKS TO GO"
620   PRINT USING 680;"----","----------","-----------"
630 DOEND
640 ELSE DO
650   PRINT USING 680;"DATE","DAYS TO GO","WEEKS TO GO","PERCENT"
660   PRINT USING 680;"----","----------","-----------","-------"
670 DOEND
680 IMAGE 15X,4A,14X,10A,3X,11A,5X,7A
690 FOR I1=S1 TO F1
700   A1=I1
710   GOSUB 1000
720   GOSUB 3000
730   IF (F1-I1)//7<>INT((F1-I1)/7) THEN 770
740   W1=W1-1
750   W2=1
760   GOTO 780
770   W2=0
780   IF D2=1 THEN PRINT " "
790   T1=((I1-S1)/(F1-S1)*100)
800   IF A1$="N" AND W2=0 THEN PRINT USING 840;D1$(D2),M1$(M1),D1,Y1,F1-I1
810   IF A1$="N" AND W2=1 THEN PRINT USING 850;D1$(D2),M1$(M1),D1,Y1,F1-I1,W1
820   IF A1$="Y" AND W2=0 THEN PRINT USING 860;D1$(D2),M1$(M1),D1,Y1,F1-I1,T1
830   IF A1$="Y" AND W2=1 THEN PRINT USING 870;D1$(D2),M1$(M1),D1,Y1,F1-I1,W1,T1
840   IMAGE 9A,2X,9A,X,DD,",",X,DDDD,6X,DDDDD
850   IMAGE 9A,2X,9A,X,DD,",",X,DDDD,6X,DDDD,8X,DDDD
860   IMAGE 9A,2X,9A,X,DD,",",X,DDDD,6X,DDDDD,23X,DDD
870   IMAGE 9A,2X,9A,X,DD,",",X,DDDD,6X,DDDDD,8X,DDDD,11X,DDD
880 NEXT I1
890 STOP
900 DATA "SUNDAY","MONDAY","TUESDAY","WEDNESDAY"
910 DATA "THURSDAY","FRIDAY","SATURDAY"
920 DATA "JANUARY","FEBRUARY","MARCH","APRIL","MAY","JUNE","JULY"
930 DATA "AUGUST","SEPTEMBER","OCTOBER","NOVEMBER","DECEMBER"
1000 REM*****************************************************
1010 REM*                                                   *
1020 REM* ROUTINE TO CALCULATE DAY, MONTH AND YEAR, GIVEN   *
1030 REM* THE DAY OF THE CENTURY.                           *
1040 REM*                                                   *
1050 REM* VALUES PASSED: A1 - DAY OF CENTURY.               *
1060 REM*                                                   *
1070 REM* VALUES RETURNED: D1 - DAY                         *
1080 REM*                  M1 - MONTH                       *
1090 REM*                  Y1 - YEAR                        *
1100 REM*                                                   *
1110 REM* LOCAL VARIABLES: X0                               *
1120 REM*                                                   *
1130 REM*****************************************************
1140 X0=(4*(A1-59))-1
1150 Y1=INT(X0/1461)
1160 D1=INT((X0-(1461*Y1)+4)/4)
1170 M1=INT((5*D1-3)/153)
1180 D1=INT((5*D1-3-153*M1)/5)+1
1190 IF (M1<10) THEN 1240
1200 M1=M1-9
1210 Y1=Y1+1
1220 Y1=Y1+1900
1230 GOTO 1260
1240 M1=M1+3
1250 Y1=Y1+1900
1260 RETURN
2000 REM*****************************************************************
2010 REM*                                                               *
2020 REM* ROUTINE TO CALCULATE THE DAY OF THE CENTURY FROM THE DAY,     *
2030 REM* MONTH AND YEAR.                                               *
2040 REM*                                                               *
2050 REM* VALUES PASSED: D1 - DAY                                       *
2060 REM*                  M1 - MONTH                                   *
2070 REM*                  Y1 - YEAR                                    *
2080 REM*                                                               *
2090 REM* VALUE RETURNED: A1 - DAY OF THE CENTURY                       *
2100 REM*                                                               *
2110 REM* LOCAL VARIABLES: X0, X1, X2, X3                              *
2120 REM*                                                               *
2130 REM*****************************************************************
2140 IF M1>2 THEN 2190
2150 X3=M1+9
2160 X0=Y1-1
2170 X0=X0-1900
2180 GOTO 2210
2190 X3=M1-3
2200 X0=Y1-1900
2210 X1=INT((1461*X0)/4)
2220 X2=INT((153*X3+2)/5)
2230 A1=X1+X2+D1+59
2240 RETURN
3000 REM*****************************************************************
3010 REM*                                                               *
3020 REM* ROUTINE TO CALCULATE THE DAY OF THE WEEK FROM THE DAY,        *
3030 REM* MONTH AND YEAR.                                               *
3040 REM*                                                               *
3050 REM* VALUES PASSED: D1 - DAY                                       *
3060 REM*                  M1 - MONTH                                   *
3070 REM*                  Y1 - YEAR                                    *
3080 REM*                                                               *
3090 REM* VALUE RETURNED: D2 - DAY OF WEEK, WHERE 1=SUNDAY, ETC.       *
3100 REM*                                                               *
3110 REM* LOCAL VARIABLES: X0, X1, X2, X3                              *
3120 REM*                                                               *
3130 REM*****************************************************************
3140 X1=Y1-1900
3150 IF M1>2 THEN 3190
3160 X2=M1+10
3170 X1=X1-1
3180 GOTO 3200
3190 X2=M1-2
3200 X3=INT((2.6*X2)-.19999)
3210 X0=INT(X1/4)
3220 D2=X3+D1+X1+X0-34
3230 IF D2>=1 THEN 3260
3240 D2=D2+7
3250 GOTO 3230
3260 IF D2>6 THEN 3290
3270 D2=D2+1
3280 RETURN
3290 D2=D2-7
3300 GOTO 3260
```

```
>RUN
DAYSTOGO

**** DAYS TO GO ****

Instructions? type YES or NO:NO

Enter start date - DD,MM,YYYY:24,12,1983

Enter finish date - DD,MM,YYYY:11,1,1984

Print percentages?:YES
```

DATE			DAYS TO GO	WEEKS TO GO	PERCENT
SATURDAY	DECEMBER	24, 1983	18		0
SUNDAY	DECEMBER	25, 1983	17		6
MONDAY	DECEMBER	26, 1983	16		11
TUESDAY	DECEMBER	27, 1983	15		17
WEDNESDAY	DECEMBER	28, 1983	14	2	22
THURSDAY	DECEMBER	29, 1983	13		28
FRIDAY	DECEMBER	30, 1983	12		33
SATURDAY	DECEMBER	31, 1983	11		39
SUNDAY	JANUARY	1, 1984	10		44
MONDAY	JANUARY	2, 1984	9		50
TUESDAY	JANUARY	3, 1984	8		56
WEDNESDAY	JANUARY	4, 1984	7	1	61
THURSDAY	JANUARY	5, 1984	6		67
FRIDAY	JANUARY	6, 1984	5		72
SATURDAY	JANUARY	7, 1984	4		78
SUNDAY	JANUARY	8, 1984	3		83
MONDAY	JANUARY	9, 1984	2		89
TUESDAY	JANUARY	10, 1984	1		94
WEDNESDAY	JANUARY	11, 1984	0	0	100

PROGRAMS FOR VIC-20

BLOCKADE

Blockade is a two player game that demands sharp reflexes and skill.

It is a game like the light cycle scene in 'TRON'. The object of the game is to make your opponent crash into either the wall, his own trail or your trail.

Shaun Clarke
Henley Beach SA

```
0 POKE36879,9:GOSUB41:REM (C) SHAUN CLARK...
1 PRINT"E":POKE36879,9
2 GOSUB36
3 A=1:B=-1
4 X=7727:Y=8138
5 GETA$:IFA$=""THEN14
6 IFA$="W"THENA=-22
7 IFA$="@"THENB=-22
8 IFA$="A"THENA=-1
9 IFA$=":"THENB=-1
10 IFA$="S"THENA= 1
11 IFA$=";"THENB= 1
12 IFA$="Z"THENA=22
13 IFA$="/"THENB=22
14 IFPEEK(X+A)=102THEN26
15 IFPEEK(X+A)=160THEN26
16 IFPEEK(Y+B)=160THEN31
17 IFPEEK(Y+B)=102THEN31
18 IFPEEK(X+A)=86THEN26
19 IFPEEK(Y+B)=86THEN31
20 FORL=1TO100:NEXTL
21 X=X+A:Y=Y+B
22 POKEX,160
23 POKEY,102
24 POKE36878,15:POKE36876,180:FORT=1TO150:NEXT:POKE36878,0
25 GOTO5
26 POKE36878,15:POKE36876,240:FORL=1TO700:NEXT:POKE36876,0
27 PRINT"♥QQQQQQQQQ]]]]]]]]]▒ WON":FORT=1TO2000:NEXT
28 RS=RS+1
29 IFRS>=10THEN65
30 GOTO60
31 POKE36878,15:POKE36876,240:FORL=1TO700:NEXT:POKE36876,0
32 PRINT"♥QQQQQQQQQ]]]]]]]]]R ■ WON":FORT=1TO2000:NEXT
33 LS=LS+1
34 IFLS>=10THEN62
35 GOTO60
36 PRINT"♥xxxxxxxxxxxxxxxxxxxxxx";
37 FORT=1TO21:PRINT"x                    x";:NEXT
38 PRINT"xxxxxxxxxxxxxxxxxxxxxx";
39 POKE8185,86:POKE8184,86
40 RETURN
41 PRINT"♥QQQQQQQ]]]]]]]ERBLOCKADE■"
42 PRINT"Q]]INSTRUCTIONS (Y/N)"
43 GETS$:IFS$=""THEN43
44 IFS$="Y"THEN47
45 IFS$="N"THENRETURN
46 GOTO43
47 PRINT"♥THIS IS A TWO PLAYER  GAME USING SHARP RE-  FLEXES AND SKILLS."
48 PRINT"R ■ IS IN THE TOP LEFT  HAND CORNER AND ▒ IS  IN THE BOTTOM RIGHT."
49 PRINT"THE OBJECT OF THE GAMEIS TO TRY TO MAKE YOUROPPONITE CRASH INTO A"
50 PRINT"WALL OR INTO A TRAIL"
51 PRINT"Q]]]]]]ANY KEY"
52 REM THIS IS FOR TWO PLAYERS
53 GETR$:IFR$=""THEN53
54 PRINT"♥THE MOVEMENT KEYS ARE"
55 PRINT"QW-UP A-LEFT S-RIGHT   Z-DOWN FOR THE TOP (R ■) PLAYER."
56 PRINT"WG-UP :-LEFT ;-RIGHT   /-DOWN FOR THE BOTTOM  (▒) PLAYER."
57 PRINT"QQ]]]]]]ANY KEY"
58 GETR$:IFR$=""THEN58
59 RETURN
60 PRINT"♥QQQQQ]]]]]▒ SCORE IS "RS
61 PRINT"QQQQQ]]]]]R ■ SCORE IS "LS:FORT=1TO2000:NEXT:PRINT"♥":GOTO1
62 PRINT"♥QQQTHIS COMPETITION  WAS FOR THE FIRST PLAYER  TO GET TO 10 PTS AND  "
63 PRINT"R ■ GOT THERE FIRST SO       HE **WON**"
64 GOTO68
65 PRINT"♥QQQTHIS COMPETITION  WAS FOR THE FIRST PLAYER  TO GET TO 10 PTS AND  "
66 PRINT"▒ GOT THERE FIRST SO        HE **WON**"
67 GOTO68
68 PRINT"QQ]]ANOTHER GAME (Y/N)"
69 GETF$:IFF$=""THEN69
70 IFF$="Y"THENRUN
71 END
```

MONOPOLY MANAGER

Here is a program that is not actually a game, rather it assists in playing the traditional board game of Monopoly. The screen displays the names and bank balances of up to six players and rolls the dice on screen when you hit the space bar.

After entering the program enter the name of each player. Then, roll the dice. Finally select which transaction you require and enter the requested information. The computer does the rest. If a player ends up with no money then he is declared bankrupt and no more data will be accepted for him.

If you wish to give each player more money to begin with alter the value of BB(N) in line 95.

B.W. Madden
Surry Hills NSW

```
10 PRINT"     MONOPOLY MANAGER   "
20 FORT=1TO500:NEXT
30 PRINT"        B.W.MADDEN      "
40 PRINT"      FEBRUARY 1983    "
50 FORT=1TO1000:NEXT:PRINT"   INSTRUCTIONS (Y/N)"
60 GETA$:IFA$<>"Y"ANDA$<>"N"THEN60
65 IFA$="N"THEN80
70 PRINT"THIS PROGRAM ROLLS THEDIE (BY HITTING SPACE BAR) AND KEEPS A BANK BAL
ANCE ";
71 PRINT"FOR UP TO SIX PLAYERS."
72 PRINT"AFTER EACH ROLL OF THEDIE SELECT FROM THE   OPTIONS WHICH"
73 PRINT"TRANSACTION YOU WANT."
74 PRINT"THEN ENTER THE PLAYER NUMBER(S) AND THE     AMOUNT."
75 PRINT"    NOW HIT ANY KEY."
76 GETA$:IFA$=""THEN76
80 PRINT"HOW MANY PLAYERS(2-6)":INPUTM
85 IFM<2ORM>6THENGOTO80
90 PRINT"ENTER PLAYERS NAMES.  (MAX 8 LETTERS)."
91 FORN=1TOM
92 INPUTP$(N)
93 IFLEN(P$(N))>8THENPRINT"THAT NAME IS TOO LONG.ABBREVIATE IT.":GOTO92
94 NEXTN
95 FORN=1TOM:BB(N)=800:NEXTN:B$="BANKRUPT"
100 REM SCREEN PRINT ROUTINE
110 PRINT"MONOPOLY MANAGER"
111 FORX=8164TO8185:POKEX,102:POKEX+30720,6:NEXTX
115 PRINT"HIT SPACE TO ROLL DICE"
120 FORN=1TOM:PRINTTAB(10);""N"";P$(N):PRINTTAB(11);"";BB(N)""
121 POKE38432+N*44,1
122 IFBB(N)=>0THENGOTO123
123 IFBB(N)<0THENPRINTTAB(11)"BANKRUPT"
124 NEXTN
130 REMPRINTTRANSACTION OPTIONS
140 PRINT"**SELECT TRANSACTION**"
160 PRINT"A PLAYER PASSES GO"
170 PRINT"B BANK PAYS PLAYER"
180 PRINT"C PLAYER PAYS BANK"
185 PRINT"D PLAYER PAYS PLAYER"
186 PRINT"E NO TRANSACTION"
191 GETA$:IFA$<>" "THEN191
192 IFA$=" "THENGOSUB1000
193 GETS$:IFS$<>"A"ANDS$<>"B"ANDS$<>"C"ANDS$<>"D"ANDS$<>"E"THEN193
200 IFS$="A"THENGOSUB1950:GOTO300
201 IFS$="B"THENGOSUB1950:GOTO400
202 IFS$="C"THENGOSUB1950:GOTO500
203 IFS$="D"THENGOSUB1950:GOTO600
204 GOTO110
300 PRINT"ENTER PLAYER NUMBER":INPUTQ:IFQ<1ORQ>6THENGOTO300
305 IFBB(Q)<0THENGOSUB1500:GOTO300
310 BB(Q)=BB(Q)+200:GOTO110
400 PRINT"TO PLAYER NUMBER":INPUTQ:IFQ<1ORQ>6THENGOTO400
401 IFBB(Q)<0THENGOSUB1500:GOTO400
405 PRINT"ENTER AMOUNT        ":INPUTA
410 BB(Q)=BB(Q)+A:GOTO110
500 PRINT"FROM PLAYER NUMBER":INPUTQ:IFQ<1ORQ>6THENGOTO500
501 IFBB(Q)<0THENGOSUB1500:GOTO500
505 PRINT"ENTER AMOUNT        ":INPUTA
510 BB(Q)=BB(Q)-A:GOTO110
600 PRINT"TO PLAYER NUMBER":INPUTQ:IFQ<1ORQ>MTHENGOTO600
601 IFBB(Q)<0THENGOSUB1500:GOTO600
602 PRINT"FROM PLAYER NUMBER    ":INPUTP:IFP<1ORP>MTHENGOTO60
2
```

▷

MONOPOLY MANAGER

```
603 IFP=QTHENGOTO602
604 IFBB(P)<0THENGOSUB1510:GOTO602
610 PRINT"ENTER AMOUNT          ":INPUTA
620 BB(Q)=BB(Q)+A:BB(P)=BB(P)-A:GOTO110
1000 REM DIE ROLL/PRINT ROUTINE
1010 DO$="┌──┐Q|Q|Q|▒▒▒▒▒|Q|Q|Q|└──┘"
1019 D0$="Q   Q▒▒   Q▒▒   "
1020 D1$="QQ▒▒▒K"
1030 D2$="Q▒▒▒▒▒K"
1040 D3$="Q▒▒▒▒▒K"
1050 D4$="Q▒▒ ▒▒▒▒▒ ▒K"
1060 D5$="Q▒▒ ▒Q▒▒▒▒ ▒K"
1070 D6$="Q▒▒▒▒▒▒▒▒▒▒K"
1080 PRINT"SQQQ▒▒▒"DO$:PRINT"SQQQ▒▒▒"D0$
1090 A=INT(RND(1)*6+1)
1110 IFA=1THENPRINT"SQQQ▒▒▒"D1$
1120 IFA=2THENPRINT"SQQQ▒▒▒"D2$
1130 IFA=3THENPRINT"SQQQ▒▒▒"D3$
1140 IFA=4THENPRINT"SQQQ▒▒▒"D4$
1150 IFA=5THENPRINT"SQQQ▒▒▒"D5$
1160 IFA=6THENPRINT"SQQQ▒▒▒"D6$
1170 PRINT"SQQQQQQQQ▒▒▒"DO$:PRINT"SQQQQQQQQ▒▒▒"D0$
1180 B=INT(RND(1)*6+1)
1210 IFB=1THENPRINT"SQQQQQQQQ▒▒▒"D1$
1220 IFB=2THENPRINT"SQQQQQQQQ▒▒▒"D2$
1230 IFB=3THENPRINT"SQQQQQQQQ▒▒▒"D3$
1240 IFB=4THENPRINT"SQQQQQQQQ▒▒▒"D4$
1250 IFB=5THENPRINT"SQQQQQQQQ▒▒▒"D5$
1260 IFB=6THENPRINT"SQQQQQQQQ▒▒▒"D6$
1310 IFA<>BTHENPRINT"SQQQQQQQQQQQQQQQ          Q▒▒▒▒▒▒▒▒▒▒▒   ":RETURN
1400 PRINT"SQQQQQQQQQQQQQQQA DOUBLE!!Q▒▒▒▒▒▒▒▒▒▒ROLL AGAIN";:GOTO193
1500 PRINT"PLAYER";Q;"IS BANKRUPT":RETURN
1510 PRINT"PLAYER";P;"IS BANKRUPT":RETURN
1950 FORX=8054TO8185:POKEX,32:NEXTX:RETURN
```

ORIENTEER

Lost in the mystic northern forest!

Escape lies to the south but each step weakens you and the way is guarded by wolves, fierce lions, grizzly bears and evil ogres.

Beware of disappearing trees and above all else the ancient stone rings that transport the unwary to someplace in the north.

To travel north, south, east or west you simply press the corresponding N, S, E or W key.

Mapping the terrain is best achieved on 12 by 100 square grid paper. You start at the most north-west corner.

The terrain "wraps around" west and east, as does your monitor screen, therefore travelling east 12 moves puts you one move south.

The program although short used almost all BASIC memory on the 3.5K VIC-20 through use of the DIM statement that sets an integer array to contain the terrain information.

Please note – the underlined "Q" in line 4 is a cursor down which appears as a reverse "Q" on the screen.

Peter Bagust
Sans Souci NSW

```
1 L$="TREECAVEROCKHILLWOODRINGOGREBEARLI
ONWOLF":DIMR%(1200):INPUT"NAME";N$:S=999
2 A=PEEK(197):ON-(A=64)GOTO2:S=S-1:R%(R)
=L:FORT=0TO350:NEXT:IFR>1188THEN6
3 R=R+(A=9)-(A=49)+12*((A=28)-((A=41))):
R=R*-(R>-1):L=R%(R):IFL=0THENL=INT(RND(1
)*10)
4 PRINT"QLANDMARK "MID$(L$,(L*4+1),4):R=
R-INT(RND(1)*R*-(L=5)):IFL<6THEN2
5 ON-(RND(1)<(L/10))GOTO2:PRINTN$" WAS A
TTACKED":S=S-15+L:PRINT"STRENGTH"S:IFS>0
THEN2
6 PRINT"THY JOURNEY ENDETH":IFS>0THENPRI
NTN$" SURVIVED! WILL THE NEXT":RUN
```

HI-RES SCREEN MACHINE CODE LOADER

This program sets up a 160 by 160 hi-res screen and incorporates a routine for plotting points on the screen. It is available for use with either BASIC or machine language routines. A 3K expander or Super Expander must be in place for use ones own basic programs.

To use the routines type in the program, run it, and then type NEW. The machine code is now in place beginning at 7296 (take care when typing in the data statements as an error here may cause the program to crash later on). Basic programs can now be typed in as usual. Alternatively the above program could be incorporated inside a BASIC program.

To set up the hi-res screen use SYS 7296.

To plot points poke an X value into 7679 and a Y value into 7678 then use SYS 7394.

The value in 7677 determines whether the point is plotted or unplotted. A zero here causes the point to be unplotted. A non-zero causes the point to be plotted. To clear whole screen use SYS 7361.

Note. If the points are to be plotted in a colour other than white then the screen must be filled with this colour after hi-res has been set up.

An example of how the routines might be used is given in the following program which plots the polar co-ordinates graph...R = SIN5J

```
10 FORJ=7296TO7465:READA:POKEJ,A:NEXT:POKE56,16:POKE55,0:POKE52,16:POKE51,0
20 DATA169,14,141,0,144,169,43,141,1,144,169,148,141,2,144,169,21,141,3,144
30 DATA169,252,141,5,144,169,147,32,210,255,169,30,133,2,169,0,133,1,160,0
40 DATA162,0,138,145,1,232,152,24,105,20,168,201,200,208,243,160,0,230,1,165
50 DATA1,201,20,208,233,169,16,133,2,169,0,133,1,160,0,169,0,145,1,230
60 DATA1,208,2,230,2,165,2,201,28,208,240,165,1,201,128,208,234,96,173,255
70 DATA29,41,248,133,1,169,16,133,2,169,0,160,20,24,101,1,144,2,230,2
80 DATA136,208,246,133,1,172,254,29,173,255,29,41,7,170,169,128,224,0,240,4
90 DATA74,202,208,252,174,253,29,208,14,141,252,29,169,255,56,237,252,29,49,
100 DATA145,1,96,17,1,145,1,96,0,0
```

The next program demonstrates how a machine code program might use the routines. To run it load and run the hi-res screen program. Then load and run this program. Type SYS 7621 and a small cross will be "bounced" around the screen in smooth hi-res motion.

This program can be run on an unexpanded VIC but any BASIC programs in memory will be lost. (POKE56,16 and POKE52,16) statements will have to be changed to (POKE56,28 and POKE52,28) in the screen program.

Wayne Rochester
Kalgoorlie WA

```
10 FORJ=7464TO7626:READA:POKEJ,A:NEXT:POKE36879,136:PRINTCHR$(5)
20 DATA162,0,173,241,29,24,125,187,29,141,255,29,232,173,240,29,125,187,29,141
30 DATA254,29,138,72,32,226,28,104,170,232,224,10,208,224,96,162,0,160,0,169
40 DATA0,141,242,29,141,243,29,142,241,29,140,240,29,138,72,152,72,169,1,141
50 DATA253,29,32,40,29,160,0,162,0,232,208,253,200,192,8,208,246,169,0,141
60 DATA253,29,32,40,29,104,168,104,170,173,242,29,208,11,232,224,154,208,14,238
70 DATA242,29,76,153,29,202,224,0,208,3,206,242,29,173,243,29,208,11,200,192
80 DATA149,208,3,238,243,29,76,87,29,136,192,0,208,3,206,243,29,76,87,29
90 DATA0,206,243,29,76,87,29,0,2,2,0,2,2,2,4,4,2,32,128,28,76,75,29
```

GALAXIAN 2

In this program the player has to shoot the alien before it invades. The player only gets one man but there are 100 possible levels (it is best to start at around level 5). As your score increases so does your level and it gets increasingly harder.

The alien will show itself for a random time (this decreases as the level goes up). The random statement in line 580 works out the cursor position for each stage down the screen. The random statement in line 620 keeps on generating numbers until it gets one lower than the present level. Between numbers being generated the player can enter commands through the keyboard.

After an alien is shot, a new alien replaces it at the top of the screen.

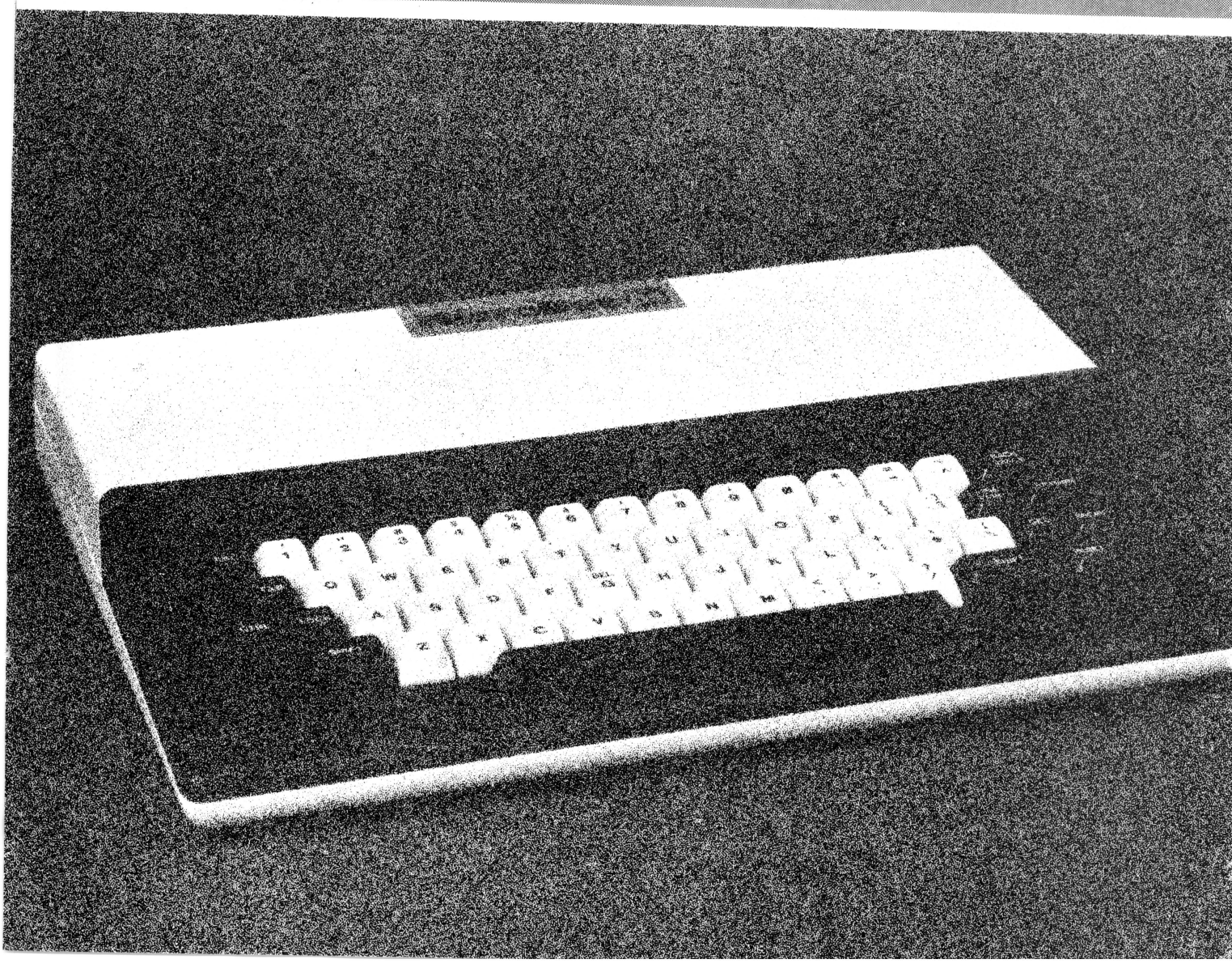
The sound effects and PCG data were POKEd into memory.

The keys to use are ',' for Left, '.' for Right and 'Z' to Fire. Also to make the game easier and to stop the keyboard from getting too badly bashed I added the keys 'A' and " " to move to the far left and right of the screen respectively. The game goes on until the alien invades.

Peter Lutton
Huon VIC

```
00100 REM * * *  G A L A X I A N   2  * * *
00110 REM * * *  By P.J.Lutton (C) 30/9/1983  * * *
00120 G=0
00130 CLS
00140 INVERSE:NORMAL
00150 CURS 25,1:PRINT"Welcome to ...."
00160 CURS 15,2:UNDERLINE:PRINT" * * *  G A L A X I A N   2  * * * "
00170 NORMAL
00180 PRINT
00190 PRINT,"Galaxian 2 is real time space action game involving"
00200 PRINT,"quick reflexes and marksman like shooting skill. The"
00210 PRINT,"object of the game is to shoot  the  leering  aliens"
00220 PRINT,"before they get to your level. You have only one man"
00230 PRINT,"so beware !!!  The control keys are ',' to go  left,"
00240 PRINT,"'.' to go right, 'Z' to fire your lasers, 'A' to go"
00250 PRINT,"to the far left of the screen and ' '  to go to the"
00260 PRINT,"far right of the screen."
00270 PRINT:PRINT,"Which skill level, 1 to 10 (1 is the easiest) ":
00280 INPUT H
00290 CURS 16,14:UNDERLINE:PRINT" * * *  G O O D  L U C K  * * * ":NORMAL:PLAY 0
,10
00300 CLS:CURS 532:PRINT"Please wait ...."
00310  F=63488+65*16
00320 FOR A=F TO F+16*10-1
00330 READ B:POKE A,B
00340 NEXT A
00350 DATA 0,0,0,31,32,67,131,128,136,68,35,16,8,7,0,0
00360 DATA 24,24,24,255,0,195,195,0,60,0,255,0,0,255,24,60
00370 DATA 0,0,0,248,4,194,193,1,17,34,196,8,16,224,0,0
00380 DATA 0,0,0,0,1,2,4,8,16,32,64,128,128,128,128,255
00390 DATA 0,0,0,0,255,0,0,31,32,67,7,32,31,0,0,255
00400 DATA 24,24,60,126,255,24,24,255,24,219,219,0,255,0,0,255
00410 DATA 0,0,0,0,255,0,0,240,8,196,196,8,240,0,0,255
00420 DATA 0,0,0,0,128,64,32,16,8,4,2,1,1,1,1,255
00430 DATA 24,24,24,24,24,24,24,24,0,0,0,0,0,0,0,0
00440 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
00450 DATA 245,229,105,96,76,69,62,64,211,2,16,254,69,62,0
00460 DATA 211,2,16,254,13,32,239,225,241,201
00470 FOR F=15000 TO 15024
00480 READ I:POKE F,I:NEXT F
00490 CURS 16,14:PRINT"*** Push any key to continue ***"
00500 A4$=KEY:IF A4$="" THEN 500
00510 CLS
00520 PCG
00530 C=32:CURS C,15:PRINT"DEFGH"
00540 GOSUB 1000
00550 S=0
00560 S=S+1
00570 REM IF S=15 THEN NORMAL:GOTO 13800
00580 B=INT(RND*56):IF B<5 THEN 580
00590 CURS B,S:PRINT"ABCABC"
00600 IF S=15 THEN NORMAL:GOTO 930
00610 A1$=KEY
00620 E=INT(RND*100)
00630 IF E<H THEN CURS B,S:PRINT"JJJJJJ":GOTO 560
00640 IF G=200 OR G=600 OR G=1080 OR G=1660 OR G=2220 OR G=2980 OR G=3820 OR G=4
760 OR G=5780 OR G=6900 THEN LET H=H+1:G=G+100*H:GOSUB 1000
00650 IF C=2 THEN CURS C,15:PRINT"JJJJJ":C=5:CURS C,15:PRINT"DEFGH":NORMAL:CURS
1,15:PRINT G:PCG
00660 IF C=59 THEN CURS C,15:PRINT"JJJJJ":C=56:CURS C,15:PRINT"DEFGH"
00670 IF A1$="," THEN CURS C,15:PRINT"JJJJJ":C=C-3
00680 IF A1$="." THEN CURS C,15:PRINT"JJJJJ":C=C+3
00690 IF A1$="Z" OR A1$="z" THEN GOTO 740
00700 IF A1$="A" OR A1$="a" THEN CURS C,15:PRINT"JJJJJ":C=5:CURS C,15:PRINT"DEFG
H"
00710 IF A1$=" " THEN CURS C,15:PRINT"JJJJJ":C=56:CURS C,15:PRINT"DEFGH"
00720 CURS C,15:PRINT"DEFGH"
00730 GOTO 610
00740 D=15
00750 FOR M=1 TO 5:OUT 2,248:OUT 2,184:NEXT M
00760 D=D-1
00770 IF D=0 THEN 800
00780 CURS C+2,D:PRINT"I"
00790 GOTO 760
00800 D=15
00810 D=D-1
00820 IF A1$=":" THEN CURS C,15:PRINT"JJJJJ":C=56:CURS C,15:PRINT"DEFGH"
00830 IF D=0 THEN 860
00840 CURS C+2,D:PRINT"J"
00850 GOTO 810
00860 IF C+2=B OR C+2=B+1 OR C+2=B+2 OR C+2=B+3 OR C+2=B+4 OR C+2=B+5 THEN 870 E
LSE 610
00870 G=G+20
00880 CURS B,S:PRINT"JJJJJJ"
00890 CURS 1,14:NORMAL:PRINT"SCORE"
00900 CURS 1,15:PRINT G:PCG
00910 FOR F=4097 TO 4150:L=USR(15000,F):NEXT F
00920 GOTO 550
00930 FOR F=4095 TO 3847 STEP -1:L=USR(15000,F):NEXT F
00940 CURS 522:PRINT"*  *  *   G A M E  O V E R    *  *  *"
00950 CURS 28,10:PRINT"SCORE : ";G
00960 CURS 13,11:PRINT"WOULD YOU LIKE ANOTHER GAME (Y OR N) ?"
00970 A5$=KEY:IF A5$="" THEN 970
00980 IF A5$="Y" THEN INVERSE:NORMAL:CLEAR:RESTORE:GOTO 270
00990 CLS:END
01000 NORMAL:CURS 1,4:PRINT"LEVEL"
01010 CURS 1,5:PRINT H
01020 CURS 1,14:PRINT"SCORE"
01030 CURS 1,15:PRINT G:PCG
01040 RETURN
```

PROGRAMS FOR MICROBEE



SOLITAIRE

```
00100CLS:CURS20,8:PRINT"DO YOUWANT INSTRUCTIONS ":INPUTA3$
00110IFA3$="Y"ORA3$="y"THENGOSUB910
00120POKE220,121
00130DIMP(16):DIMM(16):DIMO(16):DIMQ(16)
00140CLS:RESTORE
00150CURS20,4:PRINT"1"
00160CURS19,5:PRINT"/ \"
00170CURS18,6:PRINT"2 - 3"
00180CURS17,7:PRINT"/ \ / \"
00190CURS16,8:PRINT"4 - 5 - 6"
00200CURS15,9:PRINT"/ \ / \ / \"
00210CURS14,10:PRINT"7 - 8 - 9 - 10"
00220CURS13,11:PRINT"/ \ / \ / \ / \"
00230CURS12,12:PRINT"11- 12- 13- 14- 15"
00240FORN=1TO15
00250READA,B
00260P(N)=A:O(N)=B
00270NEXTN
00280DATA20,4,18,6,22,6,16,8,20,8
00290DATA24,8,14,10,18,10,22,10,26,10
00300DATA 12,12,16,12,20,12,24,12,28,12
00310FORN=1TO15
00320Q(N)=1
00330NEXTN
00340CURS0:INPUT"WHICH ONE DO YOU WISH TO REMOVE",F
00350IFF>15ORF<1THEN340
00360Q(F)=0:CURSP(F),O(F):PRINT"* "
00370DATA1,2,4,1,3,6,2,4,7,2,5,9,3,5,8,3,6,10
00380DATA4,2,1,4,7,11,4,8,13,4,5,6,5,9,14,5,8,12
00390DATA6,9,13,6,5,4,6,10,15,6,3,1,7,4,2,7,8,9
00400DATA8,5,3,8,9,10,11,7,4,11,12,13,9,5,2,9,8,7
00410DATA10,6,3,10,9,8,12,8,5,12,13,14,13,12,11
00420DATA13,14,15,13,9,6,13,8,4,14,9,5,14,13,12,15,14,13
00430DATA15,10,6,0,0,0
00440FORN=1TO15
00450M(N)=N
00460NEXTN
00470CURS0:PRINT"
"
00480PLAY8:CURS0:PRINT"                ":CURS0:INPUT"MOVE FROM ";A
00490IFQ(A)=0THEN640
00500PLAY8:CURS1,2:PRINT"                  ":CURS1,2:INPUT"JUMP OVER ";B
00510IFQ(B)=0THEN640
00520RESTORE370:CURS1,3:PRINT"                    "
00530W=0
00540FORN=1TO37
00550IFW=2THEN590
00560IFW=2THEN660
00570READX,Y,Z
00580IFA=XTHEN600
00590NEXTN:IFW=2THEN660ELSEGOTO640
00600IFB=YTHEN620
00610NEXTN:GOTO640
00620IFQ(Z)=0THENLETW=2
00630NEXTN
00640CURS1,3
00650PRINT"BAD MOVE - TRY AGAIN":GOTO480
00660Q(X)=0:CURSP(X),O(X):PRINT"* "
00670Q(Y)=0:CURSP(Y),O(Y):PRINT"* "
00680Q(Z)=1:CURSP(Z)-1,O(Z):PRINTM(Z):W=0
00690FORN=1TO15
00700IFQ(N)=1THEN720
00710NEXTN:GOTO780
00720RESTORE370
00730FORV=1TO37
00740READX,Y,Z
00750IFN=XANDQ(Y)=1ANDQ(Z)=0THENLETW=1
00760NEXTV
00770NEXTN
00780IFW=1THEN480
00790S=0
00800FORN=1TO15
00810IFQ(N)=1THENLETS=S+1
00820NEXTN
00830IFS=1THENLETA1$="GRADE A - GENIOUS"
00840IFS=2THENLETA1$="GRADE B - VERY INTELIGENT"
00850IFS=3THENLETA1$="GRADE C - AVERAGE"
00860IFS=4THENLETA1$="GRADE D - BELOW AVERAGE"
00870IFS>4THENLETA1$="GRADE E AND BELLOW - TERRIBLE"
00880CLS
00890CURS20,8:PRINTA1$
00900FORN=1TO1000:NEXTN:GOTO140
00910 CLS
00920 UNDERLINE:CURS20:PRINT"INSTRUCTIONS":NORMAL
00930 PRINT"   THIS IS A GAME OF SOLITARE USING A TRIANGULAR GRID OF 15 "
00940 PRINT"POSITIONS. EACH POSITION HAS ITS OWN IDENTIFYING NUMBER. TO"
00950 PRINT"MOVE YOU MUST INPUT THE NUMBER OF THE PIECE YOU WISH TO MOVE "
00960 PRINT"AND THE NUMBER OF THEN PIECE YOU WISH TO JUMP OVER .FOR YOUR"
00970 PRINT"MOVE TO BE LEGAL THE HOLEON THE OTHER SIDE OF THE PIN YOU WISH"
00980 PRINT"TO JUMP OVER MUST BE MARKED BY AN ASTERIX. AT THE BEGINNING OF  THE
GAME YOU MAY REMOVE ONE PIECE AS A STARTER.  THE AIM OF THE"
00990 PRINT"GAME IS TO REMOVE AS MANY PIECES AS POSSIBLE FROM THE BOARD"
01000 PRINT"          *** HIT ANY KEY TO CONTINUE ***"
01010 A1$=KEY:IFA1$=""THEN1010
01020 RETURN
```

Solitaire is a program to simulate the IQ game of the same name. You have a board in the shape of a triangle with 15 pegs. You are asked which peg you would like to move. To move a peg you must nominate the number of the peg you wish to move and the number of the peg you wish it to jump over. The hole it will land in should be marked by an asterix (showing it is empty). The aim of the game is to remove as many of the pegs as possible by jumping them with other pegs.

There is a delay of approximately 5-10 seconds while the computer checks if there are any other moves.

This program should be simple to convert to other basics knowing that:
CURS X,Y = PRINT AT or PRINT@ ;
CURSO = moves the cursor to the top left hand corner without CLSing;
POKE 220,121 = removes cursor;
PLAY 8 = plays a note;
RESTORE X = restores data read pointer to line X;
UNDERLINE = underlines the text;
NORMAL = returns output to normal after UNDERLINE.

Keith Westley
Girraween NSW

SCREEN MASTER

Screen Master is a program I wrote that allows you to copy your normal screen onto your hidden screen, copy your hidden screen onto your first screen, and swap between the two. This is a fast and easy way to store and retrieve a second screen-full of information. The demonstration program shows some other applications in BASIC programs.

Although the program loads the machine language at memory location 1050 (Decimal) onwards, it can be placed anywhere in memory. The machine language routines are broken up into separate lines:
290 and 300: screen swapping data.
310: moves the normal screen onto the hidden screen.
320: moves the hidden screen onto the normal screen.

These three sets of data can be used individually or altogether using line 280 as a selector (as shown). After running the program try listing the program, then type USR(1050,2). Don't panic! Type USR(1050,2) again.
After running the SCREEN MASTER program type in:
J=USR(1050,0) to copy the normal screen onto hidden screen. J=USR(1050,1) to copy the hidden screen onto normal screen. J=USR(1050,2) to swap the two screens.

Wayne Grant
Canterbury NSW

```
00250 REM  *  *  *  SCREEN MASTER  *  *  *  By Wayne Grant.
00260 RESTORE 280 :FOR I=0 TO 58 :READ D
00270 POKE I+1050,D :NEXT I :RETURN
00280 DATA 121,254,0,40,42,254,1,40,26
00290 DATA 33,0,240,17,0,244,62,255,1,255,3,197,78
00300 DATA 235,70,113,235,112,35,19,193,11,184,32,242,201
00310 DATA 33,0,244,17,0,240,1,0,4,237,176,201
00320 DATA 33,0,240,17,0,244,1,0,4,237,176,201,0
```

CATCH

The object of the game is to catch as many asterisks as you can, by using the square brackets as controls (left and right respectively) to control your character, being the letter "t".

This program can easily be converted to run on other computers as the BASIC used is fairly universal, and quite simple.

David Holderness
Wahroonga NSW

```
10 CLS:CURS27,8:PRINT"C A T C H":PRINT:PRINT:PRINT"((Hit a key to begin))"
20 IF KEY$ = "" THEN 20
30 POKE 220,16:POKE 257,97
40 CLS
50 CLEAR
60 M=0
70 O=50
80 S$=0
90 X=INT(RND*32)*2
100 IF X = 0 THEN 90
110 Y=1
120 L=6
130 K=32
140 CURS K,L:PRINT"T"
150 CURS X,Y:PRINT"*"
160 IF T = 0 THEN 220
170 P3$=KEY$: IF P3$="" THEN 200
180 IF P3$ <> "" THEN 290
190 CURS K,L :PRINT "T"
200 T=T+1
210 GOTO 160
220 T=0
230 CURS X,Y:PRINT "*"
240 IF Y=5 AND X <> K THEN 330
250 IF X=K AND Y=L-1 THEN 470
  260 Y=Y+1
  270 CURS X,Y:PRINT "*"
280 GOTO 160
290 CURS K,L:PRINT " "
300 IF P3$ = "J" THEN LET K=K+2
310 IF P3$ = "C" THEN LET K=K-2
313 IF K> 64 THEN LET K=64
315 IF K< 2  THEN LET K=2
320 GOTO 190
330 W=.INT(RND*15)+1
340 PLAY W;W+3;W+7;W+3;W
350 F=0
360 CURS X,Y
370 PRINT "*"
380 FOR G = 1 TO 30:NEXT G
390 CURS X,Y:PRINT " "
400 FOR G = 1 TO 30:NEXT G
410 IF F=4 THEN GOTO 430
420 LET F=F +1:GOTO 360
   430 M=M+1:CURS 1,10:PRINT "Misses = ";M
   440 IF M = 3 THEN PLAY 0,24:GOTO 550
450 CURS 1,L:PRINT SPC(64)
460 GOTO 90
470 W=INT(RND*12)+1:PLAYW;W+12
480 S$=S$+10:IF S$=350 OR S$=500 THEN GOSUB 620
490 CURS 1,12:PRINT "Score = ";S$
500 IF S$> H$ THEN LET H$=S$
510 PRINT:PRINT "Highest score = ";H$
520 CURS 1,L:PRINT SPC(64)
530 LET O=O-1
540 GOTO 90
550 CLS
560 PRINT TAB(23);"G A M E  O V E R"
570 PRINT:PRINT"Would you like to play again ?"
580 P3$ = KEY$:IFP3$="" THEN 580
590 IF P3$="Y" THEN CLS:GOTO60
600 IF P3$="N" THEN PRINT "Bye then.":END
610 GOTO 580
620 PLAY 1;6;1;6;10;6;10;13,4
630 CURS 1,10:PRINT SPC(12)
640 FOR I = 1 TO 5
650 CURS 1,10:PRINT "Misses = 0"
660 PLAY 0,2
670 CURS 1,10:PRINT SPC(12)
680 PLAY 0,2
690 NEXT I
700 M=0:O=50
710 RETURN
```

ANOTHER CATCH!

Catch is an addictive moving graphics game. You catch the dots and the computer catches you.

Richard Larkin
Dee Why NSW

```
00100REM     By Richard Larkin

00110REM           CATCH

00120 CLS : PRINT⌿⌿TAB(7)"Welcome to Catch. In this game you have to collide wit
h"⌿"a certain number of dots. You will be chased by a computer guardthat will st
art on the right side.";

00130 PRINT " To get points you must avoid capture and catch dots. If you go off
 a side you will come back on the other side. The guard can not follow you off a
ny side."

00140 PRINT "You start in the middle. If the guard catches too many dots you can
 not fill your quota and will lose."⌿"(,)    LEFT"⌿"(.)    RIGHT"⌿"(A)    UP"⌿"(Z)
  DOWN"

00150 POKE220,63 : PRINT "The computer guard gets faster so watch out!"⌿"Any key
 to start.." : I=USR(32774)

00160 CLEAR : POKE162,30 : POKE163,128 : H1=.6

00170 SD8 : Q=INT((1.1-H1)*12) : CLS : LORES : FOR X=0TO19+Q : SET INT(RND*87+20
),INT(RND*31+8) : NEXT X

00180 J=0 : PLOT 5,3 TO 5,45 TO 121,45 TO 121,3 TO 5,3 : X=64 : Y=24 : N=0 : M=0
 : B1=100 : O1=24 : V1=0 : I1=0 : SD4

00190 CURS 22,16 : PRINT "Any key to start ...."; : I=USR(32774) : CURS 1,16 : P
RINT {A63 32};
```

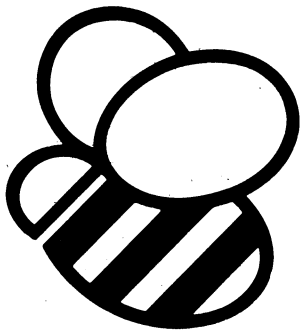
SPRACE

Sprace is a real time reflex game. Just for fun.

Richard Larkin
Dee Why NSW

```
00100REM SPRACE

00110REM By Richard Larkin

00120REM First a slow stage through a passage way.

00130REM Then speeding through space.

00140 S1=0 : NORMAL : SD8 : POKE 162,30 : POKE163,128 : POKE220,0 : CLS : SD8 :
GOSUB 330 : FOR X=-144TO-1 : READY : POKEX,Y : NEXT X : PRINT⌿"Type any key to s
tart." : I=USR(32774)

7
00150 D=2 : F=200 : C=-3872 : W1=1.9 : W=5 : CLS

00160 C1=RND*FLT(D+3) : S1=RND*FLT(D+3) : H1=FLT(D)/5 : CURS 1,15 : PRINT *A63 2
54* : CURS 33+INT(C1),15 : PRINT {A4 32} : Z1=0

00170 Z1=Z1+.1 : S1=S1+.1 : P=-3200+INT(COS(Z1)*C1+SIN(Z1)*S1+32) : POKEP-W,254
 : POKEP+W,254 : IF RND).8 THEN LET W=W+INT(RND*W1) : IF W(6ORW)8 THEN LET W1=W1*
(-1)

00180 IF F)0 THEN LET X=PEEK(256) : IF X=44 THEN POKEC,32 : C=C-1 ELSE IF X=46 T
HEN POKEC,32 : C=C+1

00190 IF RND).7 THEN POKEP+INT(RND*6-3),251

00200 POKEC,32 : PRINT : IF PEEK(C)=251 THEN OUT2,59 : OUT2,65 : F=F+10

00210 IF PEEK(C)()32 AND PEEK(C)()251 THEN 300 ELSE POKEC,255 : IF RND(4) THEN P
OKEP+INT(RND*FLT(W*2)-FLT(W)),253

00220 CURS0 : F=F-1 : PRINT F : CURS1023 : IF Z1(FLT(D*4) THEN 170

00230 PLAY6,2;2,2 : CURS 26,2 : PRINT "Well done !!" : CURS 1,15 : PRINT *A63 25
4* : CURS P+3200,15 : PRINT *A8 32* : X=0

00240 S1=S1+.1 : CURS1023 : F=F-1 : FOR X=0TO0 : POKEINT(RND*64-3200),252 : NEXT
 X : IF F)0 THEN LET X=PEEK(258) : IF X=44 THEN POKEC,32 : C=C-1 ELSE IF X=46 TH
EN POKEC,32 : C=C+1
```

```
00200 RESET X,Y : POKE257,1 : A=PEEK(258) : IF A=1 THEN LET Y=Y+1 ELSE IF A=26 T
HEN LET Y=Y-1 ELSE IF A=46 THEN LET X=X+1 ELSE IF A=44 THEN LET X=X-1

00210 IF X)119 THEN LET X=7 ELSE IF X(7 THEN LET X=119 ELSE IF Y)43 THEN LET Y=5
 ELSE IF Y(5 THEN LET Y=43

00220 Z=( NOT( POINT(X,Y))) : IF INT(B1)=X AND INT(O1)=Y THEN 290

00230 IF Z THEN 250 ELSE LET T=T+1 : S=S+T : RESET X,Y : CURS 28,16 : PRINT"SCOR
E ="S; : PLAY20 : IF T(19 THEN 250 ELSE LET J=0 : H1=H1+.05 : T=0 : IF H1)1 THEN
 LET H1=1

00240 GOTO 170

00250 SET X,Y : RESET INT(B1),INT(O1) : IF INT(B1))X THEN LET V1=-H1 ELSE IF INT
(B1)(X THEN LET V1=H1

00260 IF INT(O1))Y THEN LET I1=-H1 ELSE IF INT(O1)(Y THEN LET I1=H1

00270 B1=B1+V1 : O1=O1+I1 : IF INT(B1)=X AND INT(O1)=Y THEN 290

00280 IF POINT(INT(B1),INT(O1)) THEN LET J=J+1 : IF J=0 THEN CURS 27,8 : PRINT"I
MPOSSIBLE." : PLAY6,2;5,2;4,2;3,2;2,2;1,2 : GOTO 290 ELSE SET INT(B1),INT(O1) :
GOTO 200

00290 FOR X=1TO10 : PLAYX : K1$=KEY : NEXT X : CLS : PRINT###"You have been hit
!!"#"Total score="S+T

00300 PRINT"Any key to play again." : I=USR(32774) : GOTO 160
```



```
00250 IF RND).5 THEN POKEINT(RND*64-3200),251

00260 POKEC,32 : PRINT : IF PEEK(C)=251 THEN OUT2,59 : OUT2,65 : F=F+10

00270 IF PEEK(C)()32 AND PEEK(C)()251 THEN 300 ELSE POKEC,255

00280 K=K+1 : IF K(Q*100+100 THEN CURS0 : PRINT F : GOTO 240

00290 CURS 26,2 : PRINT "Well done !!" : PLAY0,8 : Q=Q+1 : GOTO 160

00300 FOR X=0TO9 : FOR Y=250TO247 STEP-1 : POKEC,Y : FOR Z=0TO10 : OUT2,59 : OUT
2,65 : NEXT Z : NEXT Y : NEXT X : POKEC,253 : PLAY0,10

00310 CLS : PRINT##"You have "S1*10" points." : IF INT(S1))100 THEN PLAY9;5;4;9
: PRINT"Well done !!"

00320 PRINT"Type any key to try again" : I=USR(32774) : GOTO 150

00330 CLS : PRINT##"The idea of the game is for you the "CHR(255)" to avoid the
"CHR(253)#", "CHR(254)" and "CHR(252)"; and keep your fuel positive by docking"#
"with the "CHR(251)

00340 PRINT"If you run out of fuel you will not be able to move "#"and will cras
h."#"`(' moves you left."#"`)' moves you right.":RETURN

00350 DATA 102,129,129,129,0,0,0,24,24,0,0,0,129,129,129,102,129,66,0,60,66,66,6
6,0,0,66,66,66,60,0,66,129,0,0,129,66,0,60,36,36,36,36,60,0,66,129,0,0

00360 DATA 0,0,0,0,0,66,36,24,24,36,66,0,0,0,0,0

00370 DATA 255,255,255,224,224,224,224,252,252,252,224,224,224,224,224,224

00380 DATA 129,129,66,66,36,36,24,102,102,24,36,36,66,66,129,129

00390 DATA 231,189,255,60,126,129,126,0,66,195,36,24,24,36,195,66

00400 DATA 153,126,102,219,219,102,102,153,153,102,102,219,219,102,126,153

00410 DATA 165,231,231,231,66,66,102,60,24,24,24,24,60,231,231,126
```

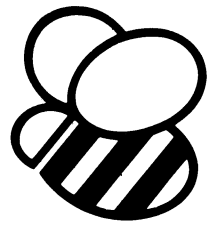
MEMSEE

The program, Memsee, is a monitor with a difference. It has no fancy commands, but it dumps 64 by 15 pieces of memory to the screen in ASCII code. It is like having a window on your computers memory, as it is continually updated. When you change a location or use certain BASIC commands you can instantly see the effect of it on the section of memory you are looking at.

Richard Larkin
Dee Why NSW

```
00100REM BY R.LARKIN

00110REM Make sure you type line 100 with at least 11 characters

00120REM after the rem statement, other wise program will not

00130REM work.

00140REM This program displays contents of memory to screen in

00150REM ASCII format. You may look at your memory and change

00160REM it in much the same way as with Moniter.

00170REM BUT as the screen is continually being up dated

00180REM you can see the effects of your changes.

00190 FOR X=2308TO2308+11 : READ Y : POKEX,Y : NEXT X : REM DELETE 110,200 after
 first RUN then save. Do not edit line 100.

00200 DATA 33,0,0,17,0,240,1,192,3,237,176,201

00210 CLS : PRINT#"To move memory pointer use 'A' and 'Z' for up and down."#"And
 ',' and '.' for left and right."#"Type (C) to change a location."#"Type (L) to
change your location"

00220 PRINT "Any key to continue.." : I=USR(32774)

00230 POKE2309,0 : POKE 2310,0 : POKE220,0 : CLS : Q=0 : W=0 : CURS 11,16 : PRIN
T "LOCATION -)  479"; : CURS 35,16 : PRINT "CONTENTS -)  "PEEK(479);

00240I=USR(2308) : POKE-3617,27 : CURS 48,16 : PRINT PEEK(L)"  "; : POKE257,1 :
POKE259,1 : K1$=KEY : IF K1$="" THEN 240

00250 IF K1$="Z" THEN LET W=W+64 ELSE IF K1$="A" THEN LET W=W-64 ELSE IF K1$="."
 THEN LET W=W-1 ELSE IF K1$="," THEN LET W=W+1

00260 IF K1$="C" THEN 300 ELSE IF K1$="L" THEN 350

00270 IF W)255 THEN LET Q=Q+1 : W=W-256

00280 IF W(0 THEN LET Q=Q-1 : W=W+256

00290 POKE2309,W : POKE2310,Q : CURS 23,16 : L=Q*256+W+479 : PRINTL"    "; : GOT
O 240

00300 A1$="" : CURS 48,16 : PRINT "/---/"{A4 8};

00310 I=USR(2308) : K1$=KEY : IF K1$)"2" OR K1$("0" THEN 310 ELSE PRINT K1$; : A
1$=A1$+K1$

00320 I=USR(2308) : K1$=KEY : IF K1$("0" OR (A1$="2"ANDK1$)"5") THEN 320 ELSE PR
INT K1$; : A1$=A1$+K1$

00330 I=USR(2308) : K1$=KEY : IF K1$("0" OR K1$)"9" THEN 330 ELSE IF A1$="25" AN
D K1$)"5" THEN 330 ELSE PRINT K1$; : A1$=A1$+K1$

00340 PLAY0 : E=INT(VAL(A1$)) : CURS 50,16 : PRINT ^A5 32+; : POKEL,E : GOTO 270

00350 A1$="" : CURS 23,16 : PRINT "/-----/"{A6 8};

00360 I=USR(2308) : K1$=KEY : IF K1$("0" OR K1$)"6" THEN 360 ELSE PRINTK1$; : A1
$=A1$+K1$

00370 I=USR(2308) : K1$=KEY : IF K1$("0" OR (A1$="6"ANDK1$)"5") THEN 370 ELSE PR
INT K1$; : A1$=A1$+K1$

00380 I=USR(2308) : K1$=KEY : IF K1$("0" OR (A1$="65"ANDK1$)"5") THEN 380 ELSE P
RINT K1$; : A1$=A1$+K1$

00390 I=USR(2308) : K1$=KEY : IF K1$("0" OR (A1$="655"ANDK1$)"3") THEN 390 ELSE
PRINTK1$; : A1$=A1$+K1$

00400 I=USR(2308) : K1$=KEY : IF K1$("0" OR (A1$="6553"ANDK1$)"5") THEN 400 ELSE
 PRINT K1$; : A1$=A1$+K1$

00410 PLAY0 : L=INT(VAL(A1$)) : Q=L/256-1 : W=L-256*(L/256)-223 : CURS 23,16 : P
RINT {A8 32}; : GOTO 270
```

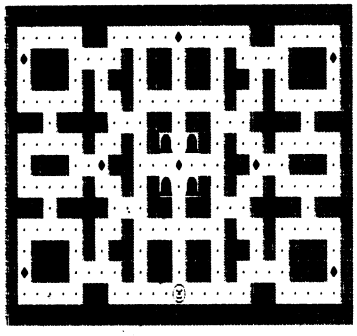
MUNCH



This game is really only suitable for Microbees with the faster clock rate modification. Even so, it gives an example of the logic behind real time games on a memory mapped screen. The DATA statements define the PCG characters used for the maze and monsters etc.

Line 430 can be altered to allow input from a joystick at the parallel port. The maze can be redefined by changing the PRINT statements in lines 260 to 380 (GG = wall, CD = dot)

C.D. Roberts
Hyde Park SA



```
00100 REM * * * MUNCH * * *
00110 CLS: POKE 220,16: CURS 23,5: PRINT " * * * MUNCH * * *"
00120 CURS 21,6: PRINT "Written for the MICROBEE"
00130 CURS 24,7: PRINT "  bv C.D.Roberts"
00140 DIM A(4),B(4),C(4),F(4,1),O(4)
00150 K=35: C0=.4: C(1)=61920: C(2)=61924: C(3)=62048: C(4)=62052
00160 RESTORE: LORES: FOR I=64528 TO 64719: READ Z: POKE I,Z: NEXT I
00170 CLS: H=0: S=0: GOSUB 210: GOSUB 250
00180 GOSUB 430: GOSUB 550: IF H THEN 750
00190 IF W THEN PLAY 10;20;10;24; FOR I=1 TO 800: NEXT I: GOSUB 210: GOSUB 250
00200 GOTO 180
00210 FOR I=1 TO 4: POKE A(I),F(I,0): POKE A(I)+1,F(I,1)
00220 A(I)=C(I): F(I,0)=199: F(I,1)=199: NEXT I
00230 E=62370: POKE E,193: POKE E+1,194: RETURN
00240 REM * * * MAZE * * *
00250 CURS 1,2: PCG: PRINT [A8 64];[A54 71]
00260 PRINT [A8 64];"GGCDCDCDCDCDGGGGCDCDCDCDCDEFCDCDCDCDCDGGGGCDCDCDCDCDGG"
00270 PRINT [A8 64];"GGEFGGGGGGCDCDCDCDGGCDGGGGCDGGGGCDGGGGCDCDCDCDCDGGGGGEFGG"
00280 PRINT [A8 64];"GGCDGGGGGGCDGGCDGGGGCDGGGGCDGGGGCDGGGGCDGGCDGGGGGGCDGG"
00290 PRINT [A8 64];"GGCDCDCDCDCDGGCDCDGGCDCDCDCDCDCDCDGGCDCDGGCDCDCDCDCDGG"
00300 PRINT [A8 64];"GGGGGGCDGGGGGGGGCDCDCDGGGGCDGGGGCDCDCDGGGGGGGGCDGGGGGG"
00310 PRINT [A8 64];"GGCDCDCDCDCDGGCDCDGGCDGGHICDHIGGCDGGCDCDGGCDCDCDCDCDGG"
00320 PRINT [A8 64];"GGCDGGGGGGCDCDEFGGGGCDCDCDEFCDCDCDGGGGEFCDCDGGGGGGCDGG"
00330 PRINT [A8 64];"GGCDCDCDCDCDGGCDCDGGCDGGHICDHIGGCDGGCDCDGGCDCDCDCDCDGG"
00340 PRINT [A8 64];"GGGGGGCDGGGGGGGGCDCDCDGGGGCDGGGGCDCDCDGGGGGGGGCDGGGGGG"
00350 PRINT [A8 64];"GGCDCDCDCDCDGGCDCDGGCDCDCDCDCDCDCDGGCDCDGGCDCDCDCDCDGG"
00360 PRINT [A8 64];"GGCDGGGGGGCDGGCDGGGGCDGGGGCDGGGGCDGGGGCDGGCDGGGGGGCDGG"
00370 PRINT [A8 64];"GGEFGGGGGGCDCDCDCDGGCDGGGGCDGGGGCDGGCDCDCDCDGGGGGEFGG"
00380 PRINT [A8 64];"GGCDCDCDCDCDGGGGCDCDCDCDCDABCDCDCDCDCDGGGGCDCDCDCDCDGG"
00390 PRINT [A8 64];[A54 71];: NORMAL: Q=186: W=0: T=1:CURS 128
00400 PRINT " Q -"\" A -"\" [ -"\" ] -": POKE 61573.204
00410 POKE 61637,11: POKE 61701,13: POKE 61765,9: PLAY 24,2: RETURN
00420 REM * * * MOVE MAN * * *
00430 M=PEEK(258): IF M=17 THEN LET D=-64 ELSE IF M=1 THEN LET D=64 ELSE IF M=27
 THEN LET D=-2 ELSE IF M=29 THEN LET D=2 ELSE 520
00440 Z=PEEK(E+D): IF Z=199 THEN 520
00450 OUT 2,64: OUT 2,0: IF Z=195 THEN LET P=10: Q=Q-1: GOTO 510
00460 IF Z=197 THEN PLAY 10,2: P=50: G=1: GOTO 510
00470 FOR I=1 TO 4: IF A(I)=E+D AND G=0 THEN NEXT* I 520
00480 IF A(I)=E+D AND G>0 THEN NEXT* I 490 ELSE NEXT I: GOTO 510
00490 P=100: PLAY 15,2: A(I)=C(I): IF F(I,0)=195 THEN LET Q=Q-1
00500 POKE A(I),202: POKE A(I)+1,203: F(I,0)=199: F(I,1)=199
00510 POKE E,32: POKE E+1,32: E=E+D: IF Q=0 THEN LET W=-1: G=0: C0=C0+.15: K=K-8
00520 POKE E,193: POKE E+1,194: S=S+P: P=0
00530 CURS 1,9: PRINT [17 S]: RETURN
00540 REM * * * MOVE MONSTERS * * *
00550 IF G>0 THEN LET G=G+1
00560 FOR I=1 TO 4: IF RND>C0 THEN LET X=A(I): GOTO 670
00570 X=A(I)-E: IF X>0 THEN LET D=-64 ELSE LET D=64
00580 V1=FRACT(FLT(E)/64): Y1=FRACT(FLT(A(I))/64)
00590 IF V1>Y1 THEN LET L=2 ELSE LET L=-2
00600 IF RND>.6 THEN LET U=L: L=D: D=U
00610 X=A(I)+D: V=PEEK(X): IF V<199 THEN 650
00620 X=A(I)+L: V=PEEK(X): IF V<199 THEN 650
00630 X=A(I)-L: V=PEEK(X): IF V<199 THEN 650
00640 X=A(I)-D: V=PEEK(X): IF V>198 THEN LET X=A(I): GOTO 670
00650 IF G=0 AND V=193 THEN POKE A(I),200: POKE A(I)+1,201: H=-1: NEXT* I 730
00660 IF G>0 AND V=193 THEN LET X=A(I)
00670 O(I)=A(I): A(I)=X: POKE O(I),F(I,0): POKE O(I)+1,F(I,1)
00680 F(I,0)=PEEK(A(I)): F(I,1)=PEEK(A(I)+1)
00690 IF G=0 THENPOKEA(I),200: POKEA(I)+1,201 ELSEPOKEA(I),202: POKEA(I)+1,203
00700 NEXT I
00710 IF G>K THEN PLAY 24
00720 IF G>K+5 THEN LET G=0: PLAY 18;22
00730 RETURN
00740 REM * * * MAN HAS BEEN CHOMPED * * *
00750 CURS 32,1: PRINT "CHOMP!": PLAY 4;8: IF T=3 THEN 790
00760 FOR I=0 TO 800: NEXT I: POKE E,32: POKE E+1,32: T=T+1: H=0
00770 CURS 32,1: PRINT [A6 32]: GOSUB 210: GOTO 180
00780 REM * * * END * * *
00790 PLAY 4;4;8: FOR I=0 TO 800: NEXT I
00800 CLS: CURS 22,7: PRINT "YOU SCORED ";S;" POINTS!"
00810 CURS 25,9: PRINT "ANOTHER GO (Y/N)?"
00820 POKE 257,1: X1$=KEY$: IF X1$="" THEN 820
00830 IF X1$<>"Y" AND X1$<>"N" THEN 820
00840 IF X1$="Y" THEN LET C0=.4: K=35: GOTO 170
00850 CLS: POKE 220,111: END
00860 DATA 0,7,24,32,64,76,133,129,131,128,72,71,32,24,7,0
00870 DATA 0,224,24,4,2,50,161,129,193,1,18,226,4,24,224,0
00880 DATA 0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0
00890 DATA 0,0,0,0,0,0,0,128,128,0,0,0,0,0,0,0
00900 DATA 0,0,0,0,1,3,7,15,15,7,3,1,0,0,0,0
00910 DATA 0,0,0,0,128,192,224,240,240,224,192,128,0,0,0,0
00920 DATA 170,85,170,85,170,85,170,85,170,85,170,85,170,85,170,85
00930 DATA 0,7,15,31,29,57,63,63,53,58,63,63,127,127,255,0
00940 DATA 0,224,240,248,184,156,252,252,172,92,252,252,254,254,255,0
00950 DATA 0,7,8,16,18,38,32,32,40,39,32,32,64,64,170,0
00960 DATA 0,224,16,8,72,100,4,4,20,228,4,4,2,2,170,0
00970 DATA 0,0,0,0,0,8,28,42,73,8,8,8,0,0,0,0
```

NOUGHTS & CROSSES

The program is based on a series of 3 grids. The first grid is Player One, the second is Player Two and the final one is the overall status. For these I used the variables Q, C and A respectively.

Firstly, the two players are asked for their names and the game continues as a normal game of 'TIC-TAC-TOE'.

The program runs from line 1001 which branches to sub-

```
LIST

  00002 POKE 220,20:POKE 140,1
  00005 LORES
  00008 REM *** WRITTEN BY ROBERT BOYCE ON 26/3/83 ***
00010 CLS:SPEED100:UNDERLINE:CURS20,8:PRINTCHR(7)"NOUGHTS AND CROSSES"CHR(7):NOR
  MAL:SPEED0
  00015 CURS 18,11:PRINT "Written by ROBERT BOYCE for the Microbee"
00020 GOSUB 500
  00025 CLS:DIM A(2,2):DIM Q(2,2):DIM C(2,2)
00050 CURS 2,8:PRINTCHR(7);"GET READY TO CHALLENGE YOUR FRIEND AT A GAME OF TIC-
  TAC-TOE":FOR X=1 TO 500:NEXTX:POKE 220,15
00051 CURS 10,10:INPUT "PLAYER 1'S NAME :";O1$
  00052 IF O1$="" THEN GOTO 51
  00053 CURS 10,11:INPUT "PLAYER 2'S NAME:";L1$:IF L1$="" THEN GOTO 53
00055 CLS:GOSUB 550
  00062 CLS
  00065 REM *** PLAYING A FRIEND ***
 00066 LORES
  00070 FOR S=0 TO 121:SET S,33:NEXTS:FOR S=0 TO 121:SET S,19:NEXTS:FOR D=47 TO 5
STEP -1:SET 40,D:NEXTD:FOR D= 47 TO 5 STEP -1:SET 81,D:NEXTD
  00074 CURS 1,15:PRINT "
    "
  00075 CURS 6,15:PRINT CHR(7);:PRINT "WHICH QUADRANT PLAYER 1(";O1$;")";:INPUT E
1$:IF E1$="" THEN GOTO 75
  00076 E1=VAL(E1$):P=74:IF E1<1 OR E1>9 THEN GOTO 74
00080 GOSUB 1000
  00090 REM **** CROSSES ****
00092 W=INT(X1):X=INT(X2):Y=INT(Y1):Z=INT(Y2)
  00095 PLOT W,Y TO X,Z:PLOT X,Y TO W,Z
 00098 IF I=9 THEN GOTO 2000
  00100 IF R4=2 THEN RETURN
 00130 CURS 1,15:PRINT "
    "
  00132 CURS 6,15:PRINT CHR(7);:PRINT "WHICH QUADRANT PLAYER 2(";L1$;")";:INPUT E
1
  00134 P=130:IF E1<1 OR E1>9 THEN GOTO 130
 00136 GOSUB 1000
  00140 REM *** NOUGHTS ***
00142 W=INT(X1):X=INT(X2):Y=INT(Y1):Z=INT(Y2)
  00145 PLOT W,Y TO W,Z:PLOT W,Y TO X,Y:PLOT X,Y TO X,Z:PLOT W,Z TO X,Z
00146 IF I=9 THEN GOTO 2000
  00148 IF T4=2 THEN RETURN
 00150 GOTO 74
  00499END
  00500 CURS 30,16:PRINT " .. PRESS ANY KEY .."
00510 IF KEY="" THEN 510
  00520 RETURN
  00550 CURS 12,1:PRINT "... RULES ..."
  00555 PRINT:PRINT "THE BOARD IS SET UP AS NORMAL INTO THE ":PRINT "NINE QUADRANT
S NUMBERED LIKE SO .."
  00560 PRINT:PRINT "  1  .  2  .  3":PRINT "  .  .  .  .  .":PRINT "  4  .  5
 6":PRINT "  .  .  .  .  .":PRINT "  7  .  8  .  9"
  00565 PRINT:PRINT "THE PLAYER GOING FIRST IS ALWAYS `CROSSES' "
00570 GOSUB 500:RETURN
  01000 REM
  01001 J=INT(E1):ON J GOTO 1010,1020,1030,1040,1050,1060,1070,1080,1090
01010 X1=3:X2=38:Y1=45:Y2=35:IF A(0,0)=1 THEN GOTO P
  01011 IF P=130 THEN 1013
 01012 Q(0,0)=1:GOTO 1014
  01013 C(0,0)=1
 01014 GOSUB 4000
  01015 A(0,0)=1:GOTO 1200
  01020 X1=43:X2=78:Y1=45:Y2=35: IF A(0,1)=1 THEN GOTO P
 01021 IF P=130 THEN 1023
  01022 Q(0,1)=1:GOTO 1024
 01023 C(0,1)=1
  01024 GOSUB 4000
 01025 A(0,1)=1:GOTO 1200
  01030 X1=84:X2=119:Y1=45:Y2=35:IF A(0,2)=1 THEN GOTO P
```

routines from the quadrant No. and then checks to see who the winner is or whether the game is a draw. The program is self-explanatory.

Robert Boyce
Mulgrave VIC

```
01031 IF P=130 THEN 1033
 01032 Q(0,2)=1 :GOTO 1034
01033 C(0,2)=1
 01034 GOSUB 4000
01035 A(0,2)=1:GOTO 1200
 01040 X1=3:X2=38:Y1=31:Y2=21:IF A(1,0)=1 THEN GOTO P
01041 IF P=130 THEN 1043
 01042 Q(1,0)=1 :GOTO 1044
01043 C(1,0)=1
 01044 GOSUB 4000
 01045 A(1,0)=1:GOTO 1200
 01050 X1=43:X2=78:Y1=31:Y2=21: IF A(1,1)=1 THEN GOTO P
01051 IF P=130 THEN 1053
 01052 Q(1,1)=1 :GOTO 1054
01053 C(1,1)=1
 01054 GOSUB 4000
01055 A(1,1)=1:GOTO 1200
 01060 X1=84:X2=119:Y1=31:Y2=21 :IF A(1,2)=1 THEN GOTO P
01061 IF P=130 THEN 1063
 01062 Q(1,2)=1 :GOTO 1064
01063 C(1,2)=1
 01064 GOSUB 4000
01065 A(1,2)=1:GOTO 1200
 01070 X1=3:X2=38:Y1=17:Y2=7:IF A(2,0)=1THEN GOTO P
01071 IF P=130 THEN 1073
 01072 Q(2,0)=1 :GOTO 1074
01073 C(2,0)=1
 01074 GOSUB 4000
01075 A(2,0)=1:GOTO 1200
 01080 X1=43:X2=78:Y1=17:Y2=7:IF A(2,1)=1 THEN GOTO P
01081 IF P=130 THEN 1083
 01082 Q(2,1)=1 :GOTO 1084
01083 C(2,1)=1
 01084 GOSUB 4000
01085 A(2,1)=1:GOTO 1200
 01090 X1=84:X2=119:Y1=17:Y2=7: IF A(2,2)=1 THEN GOTO P
01091 IF P=130 THEN 1093
 01092 Q(2,2)=1 :GOTO 1094
01093 C(2,2)=1
 01094 GOSUB 4000
01095 A(2,2)=1:GOTO 1200
 01200 I=I+1:RETURN
02000 FOR X=1 TO 1000:NEXTX:CLS
 02010 FOR X=1 TO 10:F=INT(RND*24):PLAY F,1:NEXTX
03000 CURS 16,7:PRINT "** THE GAME WAS A DRAW **"
 03010 CURS 20,8:INPUT "PLAY AGAIN (Y/N)";F1$
03020 IF F1$(1,1)="Y" THEN RUN
 03030 CLS:END
 04000 IF Q(0,0)=1 AND Q(1,0)=1 AND Q(2,0)=1 THEN GOTO 5000
04010 IF Q(0,1)=1 AND Q(1,1)=1 AND Q(2,1)=1 THEN GOTO 5000
 04020 IF Q(0,2)=1 AND Q(1,2)=1 AND Q(2,2)=1 THEN GOTO 5000
04030 IF Q(0,0)=1 AND Q(0,1)=1 AND Q(0,2)=1 THEN GOTO 5000
 04040 IF Q(1,0)=1 AND Q(1,1)=1 AND Q(1,2)=1 THEN GOTO 5000
04050 IF Q(2,0)=1 AND Q(2,1)=1 AND Q(2,2)=1 THEN GOTO 5000
 04060 IF Q(0,0)=1 AND Q(1,1)=1 AND Q(2,2)=1 THEN GOTO 5000
04070 IF Q(0,2)=1 AND Q(1,1)=1 AND Q(2,0)=1 THEN GOTO 5000
 04100 IF C(0,0)=1 AND C(1,0)=1 AND C(2,0)=1 THEN GOTO 7000
04110 IF C(0,1)=1 AND C(1,1)=1 AND C(2,1)=1 THEN GOTO 7000
 04120 IF C(0,2)=1 AND C(1,2)=1 AND C(2,2)=1 THEN GOTO 7000
04130 IF C(0,0)=1 AND C(0,1)=1 AND C(0,2)=1 THEN GOTO 7000
 04140 IF C(1,0)=1 AND C(1,1)=1 AND C(1,2)=1 THEN GOTO 7000
04150 IF C(2,0)=1 AND C(2,1)=1 AND C(2,2)=1 THEN GOTO 7000
 04160 IF C(0,0)=1 AND C(1,1)=1 AND C(2,2)=1 THEN GOTO 7000
04170 IF C(0,2)=1 AND C(1,1)=1 AND C(2,0)=1 THEN GOTO 7000
 04300 RETURN
 05000 R4=2:GOSUB 92:FOR X=1 TO 1000:NEXTX:CLS:REM *** PLAYER 1 WON ***
05005 FOR X=1 TO 20:F=INT(RND*24):PLAY F,1:NEXTX:CURS 20,7:PRINT "PLAYER ** 1 **
    ";:UNDERLINE:PRINT "WON";:NORMAL:GOTO 3010
07000 T4=2:GOSUB 142:FOR X=1 TO 1000:NEXTX:CLS: REM *** PLAYER 2 WON ***
 07005 FOR X=1 TO 20:F=INT(RND*24):PLAY F,1:NEXTX:CURS 20,7:PRINT "PLAYER ** 2 **
 ";:UNDERLINE:PRINT "WON";:NORMAL:GOTO 3010
```

FX

Pattern draws patterns on the screen depending on the values you type in, and some nice FX can be created.

Richard Larkin
Dee Why NSW

NOVEL

Novel is a program that was written just for fun. Run the program and it will create a story from the yes/no questions you ask it. If the 'Bee decides the answer is 'NO' then the computer does nothing. If the answer is 'YES' then the question is changed in syntax to become the answer. The answer is then stored as part of a string array. To display the story, enter 'print' to the question input.

The program is a good demonstration of the Microbee's string handling abilities. The major part of it is dedicated to producing syntactically correct sentences. It isn't exactly perfect, but then I never was very good at English. There are a few funny results produced and I'll leave it to the English deviated...ummm, orientated, to correct the program to process these little gloops.

The program waits for a question to be input. It then checks for a question mark at the end of the question. If that is correct, a check is made of the verb that sits at the beginning of the question. The list of verbs checked is from line 200 to 300. The list of verbs can be expanded on, just follow the format. Example:
200 A=SEARCH(AO$,"Is"):IF A=1 THEN LET AO$(;4):GOSUB "is" 420: GOTO 310

The string 'Is' is sought for in the input string. If it is in the fist character of the string then "Is"+ a space is deleted from the string. That's the LET AO$=AO$(;4). The GOSUB ('is')420 jumps to the routine to insert the verb back into the string AO$ at the correct place so the question becomes a statement.

Lines 320 to 410 do the punctuation, set the first letter of the string to capital, decide whether the answer to the question is yea or nay and if yea then stick it at the end of an array string.

The last three lines of the program insert the verb in the correct place.

Jon Barnett
Northmead NSW

```
00100 REM                **Novel**

00110 CLS:STRS(20000):S=0:L=0:DIM A2(300)
00120 A2$(0)=""
00130 PRINT"Ask Yes or No questions to create a story."
00140 INPUT"What is the question?";A0$:B=INT(RND*FLT(LEN(A0$))+1):A3$=A0$(;B,B)
00150 A=SEARCH(A0$,"print"):IF A<>0 THEN 410
00160 A=SEARCH(A0$,"PRINT"):IF A<>0 THEN 410
00170 A1$=A0$(;1,1):IF A1$=" " THEN LET A0$=A0$(;2):GOTO 170
00180 B=SEARCH(A0$,"?"):IF B=0 THEN CLS:PRINT"That wasn't a question.":GOTO 140
00190 A0$=A0$(;1,B)
00200 A=SEARCH(A0$,"Is"):IF A=1 THEN LET A0$=A0$(;4):GOSUB [" is"] 420:GOTO 310
00210 A=SEARCH(A0$,"Does"):IF A=1 THEN LET A0$=A0$(;6):GOSUB [" does"] 420:GOTO 310
00220 A=SEARCH(A0$,"Was"):IF A=1 THEN LET A0$=A0$(;5):GOSUB [" was"] 420:GOTO 310
00230 A=SEARCH(A0$,"Has"):IF A=1 THEN LET A0$=A0$(;5):GOSUB [" has"] 420:GOTO 310
00240 A=SEARCH(A0$,"Are"):IF A=1 THEN LET A0$=A0$(;5):GOSUB [" are"] 420:GOTO 310
00250 A=SEARCH(A0$,"Will"):IF A=1 THEN LET A0$=A0$(;6):GOSUB [" will"] 420:GOTO 310
00255 A=SEARCH(A0$,"Don't"):IF A=1 THEN LET A0$=A0$(;7):GOSUB [" don't"] 420:GOTO 310
00256 A=SEARCH(A0$,"Won't"):IF A=1 THEN LET A0$=A0$(;7):GOSUB [" won't"] 420:GOTO 310
00260 A=SEARCH(A0$,"Have"):IF A=1 THEN LET A0$=A0$(;6):GOSUB [" have"] 420:GOTO 310
00270 A=SEARCH(A0$,"Can"):IF A=1 THEN LET A0$=A0$(;5):GOSUB [" can"] 420:GOTO 310
00280 A=SEARCH(A0$,"Do"):IF A=1 THEN LET A0$=A0$(;4):GOSUB [""] 420:GOTO 310
00290 A=SEARCH(A0$,"Could"):IF A=1 THEN LET A0$=A0$(;7):GOSUB [" could"] 420:GOTO 310
00300 A=SEARCH(A0$,"Might"):IF A=1 THEN LET A0$=A0$(;7):GOSUB [" might"] 420:GOTO 310
00310 IF A<>1 THEN 140
00320 D=INT(RND*4):IF D=1 THEN LET A1$="!" ELSE LET A1$="."
```

```
00100REM FX

00110REM By Richard Larkin

00120REM This program is for creation of patterns on screen!

00130 CLEAR : CLS : PRINT♯♯♯"Welcome to FX. The purpose of this program is to dr
aw patterns  on the screen. All you have to do is choose a pattern type"

00140 PRINT"(1 TO 4), then type in 5 X-steps and 5 Y-steps."

00150 PRINT"I will then draw on the screen using your data."♯"Type (S) to turn m
ovement on and off."♯"Type (E) to end pattern and restart."♯"Type (C) to turn on
e frame MOD on and off."

00160 PRINT"Any other key will stop and start pattern formation."♯"Any key to co
ntinue...." : I=USR(32774)

00170 POKE220,0 : DIM X(4),Y(4)

00180 CLS : PRINT♯♯♯"PATTERN (1 TO 4)"

00190 W=INT(VAL(KEY)) : IF W(1 THEN 190

00200 PLAY10 : C1$="" : S1$="" : CLS : PRINT♯♯"  TYPE IN YOUR NUMBERS (1 TO 9)"♯
"  X  X  X  X  X    Y  Y  Y  Y  Y"

00210 FOR Z=0TO4

00220 Q=INT(VAL(KEY)) : IF Q(1 THEN 220 ELSE LET X(Z)=Q : PRINT TAB(Z*3+2)Q; : P
LAY10 : NEXT Z

00230 FOR Z=0TO4

00240 Q=INT(VAL(KEY)) : IF Q(1 THEN 240 ELSE LET Y(Z)=Q : PRINT TAB(Z*3+18)Q; :
PLAY10 : NEXT Z

00250 POKE220,63 : PRINT♯"       Any key to start ...." : I=USR(32774) : CLS : LO
RES : CURS1000 : A=0 : C=0 : U=0

00260 IF C1$="C" THEN CLS : CURS 950

00270 ON W GOTO 390,400,410,420

00280 C=C+X(A) : U=U+Y(A) : A=A+1 : IF A=5 THEN LET A=0

00290 IF C)127 THEN LET C=C-128

00300 IF U)47 THEN LET U=U-48

00310 IF S1$="S" THEN PRINT

00320 POKE257,1 : K1$=KEY : IF K1$="" THEN 260

00330 IF K1$="E" THEN 130

00340 IF K1$()"C" THEN 360 ELSE IF C1$="C" THEN LET C1$="" ELSE LET C1$="C"

00350 GOTO 260

00360 IF K1$()"S" THEN 380 ELSE IF S1$="S" THEN LET S1$="" ELSE LET S1$="S"

00370 GOTO 260

00380 IF KEY="" THEN 380 ELSE 260

00390 PLOTI C,U TO C,47-U TO 127-C,47-U TO 127-C,U TO C,U : GOTO 280

00400 PLOTI C,U TO 127-C,47-U : PLOTI C,47-U TO 127-C,U : GOTO 280

00410 PLOTI C,U TO 127-C,U TO C,47-U TO 127-C,47-U TO C,U : GOTO 280

00420 H=(47-U*2)/4 : B=(127-C*2)/4 : PLOTI C,U TO 64,U+H TO 127-C,U TO 127-C-B,2
3 TO 127-C,47-U TO 64,47-U-H TO C,47-U TO C+B,23 TO C,U : GOTO 280
```

PCG CHARACTER DESIGNER

This program is an expansion of the PCG Character designer which appeared in *Your Computer* some months ago.

Using that program, and finding the need for designing more than 1 character, I wrote this program to allow 3 characters to be designed at once, as well as adding some other features.

Because of the lack of space on the screen once the 3 – character grid has been drawn up, only one set of data can be put down the left side of the screen at a time, however this is not really seen as a problem.

One special addition to the program is a "Drawing" mode, activated by the letter "D". This automatically toggles the point when you move the cursor, making it easier to fill in large areas of the character(s). You can exit the drawing mode by pressing "D" again.

The control keys are as follows:

ESC: Move cursor up
TAB: Move cursor down
[: Move cursor left
] : Move cursor right
SPC: Toggle point
R: Reset all data
F: Set all data
I: Invert all data
H: home cursor
1,2,3: Print data for each character
SHFT + 1,2,3: Input data for each character
D: Toggles in or out of drawing mode (Shown by a "D" at top of screen)

Peter Frankenburg
Howlong NSW

```
10 REM ****************************
20 REM ** PCG Character Designer **
30 REM ** by: Peter Frankenberg. **
40 REM ** Ph: (060) - 26 5356    **
50 REM ****************************
60 REM
100 P=64528: Poke 257,2
110 FOR X=P TO P+95: Read Y: Poke X,Y: Next X
120 For X=X+1 to X+48: Poke X,0: Next X
130 For X=X+1 to X+32: Read Y: Poke X,Y: Next X
140 Goto 400
200 Rem ** Square
210 Data 0,127,64,64,64,64,64,64
220 Data 64,64,64,64,64,64,127,0
230 Data 0,254,2,2,2,2,2,2
240 Data 2,2,2,2,2,2,254,0
250 Rem ** Cross
260 Data 255,129,129,129,129,129,129,255
270 Data 255,129,129,129,129,129,129,255
280 Data 255,129,129,129,129,129,129,255
290 Data 255,129,129,129,129,129,129,255
300 Rem ** Inverse Cross
310 Data 255,254,254,254,254,254,254,128
320 Data 128,254,254,254,254,254,254,255
330 Data 255,127,127,127,127,127,127,1
340 Data 1,127,127,127,127,127,127,255
350 Rem ** Inverse Square
360 Data 0,127,127,127,127,127,127,127
370 Data 127,127,127,127,127,127,127,0
380 Data 0,254,254,254,254,254,254,254
390 Data 254,254,254,254,254,254,254,0
395 Data 128,64,32,16,8,4,2,1
400 Dim A(24,16), B(8)
500 Cls: Pcg: For X=1 to 16: For Y=16 to 62 Step 2: Curs Y,X
510 Print "AB";: Next Y: Next X
520 C=1: R=1: Curs 12,16: Print "GHI";
550 For X=1 to 8: Read B(X): Next X
600 A$=Key: If A$="" Then 600
610 If A$="[": Gosub [-1,0] 1000: If D=1: A$=" "
620 If A$="]": Gosub [1,25] 1000: If D=1: A$=" "
630 If A$=Chr(27): Gosub [-1,0] 1100: If D=1: A$=" "
640 If A$=Chr(9): Gosub [1,17] 1100: If D=1: A$=" "
650 If A$=" ": Gosub 1200
660 If A$="F": Gosub [1] 1300
670 If A$="H": Gosub 2000: R=1: C=1
680 If A$="R": Gosub [0] 1300
690 If A$>"0" And A$<"4": V=Int(Val(A$))-1: Gosub [V] 1400
700 If A$>" " And A$<"$": V=Asc(A$)-32: Gosub [V] 1700
710 If A$="I": Gosub 1800
720 If A$="D": Gosub 1900
990 Gosub 2020: Goto 600

995 Rem ** Move Cursor left & Right
1000 Var (X,Y): Gosub 2000: C=C+X: If C=Y: C=C-X
1010 Gosub 2020: Return
1095 Rem ** Move Cursor up & Down
1100 Var (X,Y): Gosub 2000: R=R+X: If R=Y: R=R-X
1110 Gosub 2020: Return
1195 Rem ** Toggle Point
1200 If A(C,R)=0: A(C,R)=1 Else Let A(C,R)=0
1210 Gosub 2020: Gosub 1500: Return
1295 Rem ** Fill with Inverse Squares
1300 Var (X): For R=1 to 16: For C=1 to 24: A(C,R)=X
1310 Gosub 2000: Next C: Next R: C=1: R=1: Gosub [X] 1600: Return
1395 Rem ** Print Data
1400 Var (W): Normal: For Y=1 to 16: V=0: Z=128
1410 For X=W*8+1 to W*8+8: If A(X,Y)=1:V=V+Z
1420 Z=Z/2: Next X: Curs 1,Y: Print W+1 ">" [I4 V];
1430 Next Y: Return
1495 Rem ** Updates PCG Character
1500 If C>8 And C<17: P=64640: X=C-8 Else If C>16: P=64656:
     X=C-16 Else Let P=64624: X=C
1510 X=B(X): P=P+R-1: If A(C,R)=0: Poke P,Peek(P)-X Else
     Poke P,Peek(P)+X
1520 Return
1595 Rem ** Update PCG after Fill or Reset
1600 Var (Y): P=64624: If Y=1: Y=255
1610 For X=P to P+47: Poke X,Y: Next X: Return
1695 Rem ** Input Data
1700 Var (W): Normal: P=64624+(W-1)*16: For V=1 to 16: R=V
1710 Curs 1,V: Print W ">"     ";: Curs 4,V:Input "" X;
1720 For Y=1 to 8: C=(W-1)*8+Y
1730 Z=(X And B(Y)): If Z=B(Y): A(C,R)=1 Else Let A(C,R)=0
1740 Gosub 2000:Next Y:Poke P,X: P=P+1: Next V: C=1: R=1:
     Return
1795 Rem ** Invert Squares
1800 For R=1 to 16: For C=1 to 24
1810 If A(C,R)=1: A(C,R)=0 Else Let A(C,R)=1
1820 Gosub 2000: Next C: Next R: C=1: R=1
1830 For X=64624 to 64671: Poke X,255-Peek(X): Next X: Return
1895 Rem ** Draw Mode
1900 Normal: If D=1: D=0: Curs 12: Print " "; Else Curs 12:
     Print "D";: D=1
1910 Return
2000 Pcg: Curs C*2+14,R: If A(C,R)=0: Print "AB"; Else
     Print "JK";
2010 Normal: Return
2020 Pcg: Curs C*2+14,R: If A(C,R)=0: Print "CD"; Else
     Print "EF";
2030 Normal: Return
```

WORM

This is a simple "Snake"-line game in Lores graphics for one person.

The object is to control a worm eating all the flashing frogs which appear on the screen. (Don't ask how the worm eats frogs, it just does!)

The only trouble is that if you crash into the wall, or yourself you are no longer alive.

You get points for just staying around, but more points for eating frogs. Eating frogs, however, causes you to grow, making the game more difficult.

The control keys for the game are:

ESC: Up
TAB: Down
[: LEFT
]: RIGHT

At the beginning of the game, the program asks for speed. This is just a number from 0-255 giving the speed of the game.

One line has been omitted, and should be added to enable the high score function.

190 POKE 62475,0 : POKE 62477,0 : DELETE 190

Peter Frankenburg
Howlong NSW

```
00001 REM  ********************************
00110 REM  ***       W  O  R  M         ***
00120 REM  *** By:  Peter Frankenberg ***
00130 REM  ***  Ph:  (060) - 26 5356  ***
00140 REM  ********************************
00150 REM
00160 REM
00170 DIM D(5000)
00175 CLS:CURS 530:GOTO 500
00180 CLS:LORES
00200 PLOT 0,0 TO 0,47 TO 100,47 TO 100,0 TO 0,0
00210 W=10:X=20:Y=10:Z=10:A=0:B=1:F=1:E=11:N=1:S1=0:G=0:P=0
00220 M=PEEK(62475)+100*PEEK(62477)
00230 PLOT W,X TO Y,Z:FOR H=1 TO 11:D(H)=10:NEXT H
00240 J=2:K=2:GOSUB 530
00250 CURS 54:PRINT"W O R M":CURS 54,2:PRINT"========="
00260 E=E+1:A1$=KEY:N=-N:P=P+1
00270 IF P>150:GOSUB 530
00280 IF E>4999:CLS:CURS 520:PRINT"C O N G R A T U L A T I O N S !":GOTO 470
00290 S1=S1+.03:IF N=-1:RESET J,K ELSE SET J,K
00300 SPEED 0:CURS 53,4:PRINT"SCORE:"INT(S1)
00310 CURS 54,6:PRINT"HIGH":CURS 53,7:PRINT"SCORE:"M:CURS 52,9:PRINT"GROWTH:"G"
";:SPEED Q:CURS 53,16:PRINT"  ";:SPEED 0
00312 CURS 52,11:PRINT"LENGTH:"E-F:CURS 52,13:PRINT"WORM:"W","X" ";:CURS 52,14:P
RINT"FROG:"J","K" ";
00320 IF A1$=CHR(27):B=1:A=0
00330 IF A1$=CHR(9):B=-1:A=0
00340 IF A1$="[":A=-1:B=0
00350 IF A1$="]":A=1:B=0
00360 D(E)=B*10+A:W=W+A:X=X+B
00370 IF W=J AND X=K:G=G+INT(RND*70)+30:GOSUB 530:S1=S1+20
00380 IF POINT(W,X)THEN 460
00390 SET W,X:IF G>0:G=G-1:GOTO 450
00400 F=F+1:RESET Y,Z
00410 IF D(F)=10:Z=Z+1
00420 IF D(F)=-10:Z=Z-1
00430 IF D(F)=1:Y=Y+1
00440 IF D(F)=-1:Y=Y-1
00450 GOTO 260
00460 SPEED 0:CURS 530:PRINT"C R A S H !":PLAY 1,10
00470 IF S1>FLT(M):CURS 580:PRINT"*** NEW HIGH SCORE ***":PLAY 20,10 ELSE 500
00480 M1$=STR(INT(S1)):POKE 62475,INT(VAL(M1(;LEN(M1$)-1)))
00490 IF LEN(M1$)>3:POKE 62477,INT(VAL(M1$(;1,LEN(M1$)-2))) ELSE POKE 62477,0
00500 INPUT"Speed?"Z7$:IFZ7$=""THEN180
00510 Q=INT(VAL(Z7$)):IF Q<0 OR Q>255 THEN 500
00520 GOTO 180
00530 P=0:RESET J,K
00540 J=INT(RND*90)+5:K=INT(RND*40)+4:IF POINT(J,K)THEN 540
00550 SET J,K:RETURN
```

SLOT MACHINE

You are in the midst of a packed casino with a 'one-arm bandit' in front of you; its jackpot steadily mounting. You insert the coin, your last one, and pull back on the handle (you really hit the / key but that spoils the fun!) and watch the shapes revolve. You hit the jackpot! Beauty! Shall you quit while you're ahead or go on gambling at the risk of going broke!

Some breath-taking sound effects and interesting characters add to the fun of this program which can become quite addictive.

Some points to note:

* The characters do not actually revolve – they just appear to do so.

* If you want a greater challenge then just increase the number of characters that appear in the game by altering lines 370, 420, 470, 540 and so on.

* You can also alter these lines to change the characters that appear by altering lines 370, 420, 470, 540 and so on.

* You can also alter these lines to change the characters that appear, if you find the ones I've chosen undesirable. To see the range of characters available, punch in: FOR X=0 TO 127:POKE 61440+(X*2),X:NEXT X:END

Anthony Lock
Mitcham VIC

```
00100 REM *** SLOT MACHINE ***
00105 REM by Anthony William Lock
00110 CLS:CURS 13,6:PRINT"How to Play slot machine..."
00120 CURS 16,7:PRINT"It costs 20c a turn -"
00130 CURS 12,8:PRINT"to insert coin, hit the / key"
00140 CURS 12,10:PRINT" To win you must match three"
00150 CURS 12,11:PRINT" objects in a horizontal row"
00160 CURS 12,12:PRINT" To start, Press the S key"
00170 CURS 12,13:PRINT"  To end, Press the E key"
00180 A1$=KEY$
00190 IF A1$="S" THEN 210
00200 GOTO 180
00210 CLS
00220 M1=2:J1=5
00230 CURS 17,1:UNDERLINE:PRINT"*** SLOT MACHINE ***"
00240 CURS 20,3:PRINT"MONEY - $";M1;"0"
00250 CURS 19,4:PRINT"JACKPOT - $";J1;"0":NORMAL
00260 CURS 13,9:PRINT"I AM READY TO ACCEPT COINS..."
00270 B1$=KEY$
00280 IF B1$="/"THEN 310
00290 IF B1$="E"THEN 970
00300 GOTO 270
00310 CLS:CURS 17,1:UNDERLINE:PRINT"*** SLOT MACHINE ***"
00320 IF M1<.2 THEN 950
00330 M1=M1-.2:J1=J1+.2
00340 CURS 20,3:PRINT"MONEY - $";M1;"0"
00350 CURS 19,4:PRINT"JACKPOT - $";J1;"0":NORMAL
00360 R=0
00370 A=INT(RND*6)+5:B=INT(RND*6)+5:C=INT(RND*6)+5
00380 POKE 61974,A:POKE 61978,B:POKE 61982,C
00390 R=R+1
00400 IF R=10 THEN 420
00410 GOTO 370
00420 B=INT(RND*6)+5:C=INT(RND*6)+5
```

MASTERMIND

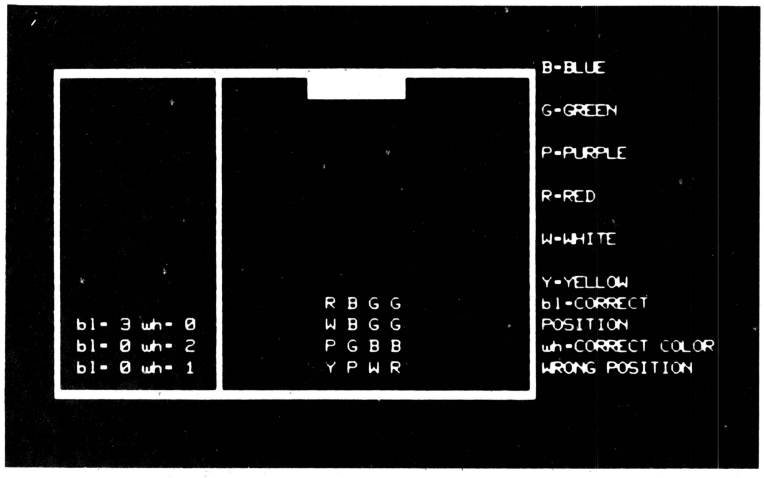
This game, written for the Microbee, is much the same as it's namesake. This version you can play by yourself or against a friend.

I will now issue a plea to anyone who can write a subroutine or program, in which the computer tries to guess the code, and send it in to this magazine. I'm afraid the maths is a bit beyond me! *(OK, Hackers the race is on. If no-one comes up with one I'll publish an algorithm in a couple of months. EM.)*

This program is very flexible for changing and converting to other computers. This is because it is only from line 350 to line 900 that the most important information is processed. To convert therefore, is mainly a matter of changing variables and designing your screen layout (see the screen dump).

I recommend that you invent P.C.G characters to replace the plain looking letters. If you are the owner of that rare creature, the ColourBee, the use of colours for the code pegs would make this game visually much better. The game uses about 5K .... GOOD LUCK!

Greg Alcock,
Oak Park, Vic




```
00010 REM***MASTERMIND BY...***
00020 REM*** GREG ADCOCK ***
00030 REM*** 20/1/1984 ***
00040 REM*** FOR PUBLIC USE !***
00040 REM*** FOR PUBLIC USE !***
00100CLS
00110POKE220,1
00120DIMU1(4):W3=0
00130 DIMG(4)
00140 DIMA1(3):DIMU(3)
00150GOTO940
00160CLS
00170 T1=0:J1=0:K=0
00180 L=0:PLAY0,20:CLS
00190 LORES
00200 PLOT 30,45 TO 88,45 TO 88,0 TO30,0TO30,45
00210 PLOT 30,0 TO0,0 TO 0,45 TO 30,45
00220 PLOT 47,43 TO64,43
00230 PLOT 47,42 TO64,42
00240 PLOT 47,44TO64,44
00250 CURS46,1:PRINT"B=BLUE"
00260 CURS46,3:PRINT"G=GREEN"
00270 CURS46,5:PRINT"P=PURPLE"
00280 CURS46,7:PRINT"R=RED"
00290 CURS46,9:PRINT"W=WHITE"
00300 CURS46,11:PRINT"Y=YELLOW"
00310 CURS46,12:PRINT"bl=CORRECT"
00320 CURS46,13:PRINT"POSITION"
00330 CURS46,14:PRINT"wh=CORRECT COLOR"
00340 CURS46,15:PRINT"WRONG POSITION"
00350 FORI=0TO3
00360 P=INT(RND*8):IFP=0ORP=1ORP=8THEN360
00370 IFP=2THENLETU(I)=82
00380 IFP=3THENLETU(I)=66
00390 IFP=4THENLETU(I)=71
00400 IFP=5THENLETU(I)=87
00410 IFP=6THENLETU(I)=89
00420 IFP=7THENLETU(I)=80
00430 A1$(I)=CHR(U(I))
00440 NEXTI
00450 T=40:S=15
00460 REM
00470 T=40:S=S-1
00480 CURS 3,16
00490L=L+1:IFL=11THEN900
00500 FORI=0TO4
00510 G(I)=0:U1$(I)=""
00520 NEXTI
00530 FORI=1TO4
00540 CURS 50+T,S
00550 K1$=KEY:IFK1$=""THEN550
00560PRINT K1$;
00570 IFK1$="R"ORK1$="B"ORK1$="G"ORK1$="W"ORK1$="Y
"ORK1$="P"THEN580 ELSE540
00580 U1$(I)=K1$
00590 PLAY 12,2
00600 T=T+2
00610 NEXTI
00620 C=0:B=0
00630FORI=0TO3
00640 IFA1$(I)=U1$(I+1)THENLETB=B+1:G(I+1)=3+1
00650NEXTI
00660 FORI=1TO4
00670 IFG(I)>1THEN 840
00680IFG(I)>1THEN 720
00690 IFF1=1THEN720
00700 IFI=1THEN720
```

```
00430 POKE 61978,B:POKE 61982,C
00440 R=R+1
00450 IF R=20 THEN 470
00460 GOTO 420
00470 C=INT(RND*6)+5
00480 POKE 61982,C
00490 R=R+1
00500 IF R=35 THEN 520
00510 GOTO 470
00520 R=0:PLAY 0,4
00540 D=INT(RND*6)+5:E=INT(RND*6)+5:F=INT(RND*6)+5
00550 POKE 62038,D:POKE 62042,E:POKE 62046,F
00560 R=R+1
00570 IF R=10 THEN 590
00580 GOTO 540
00590 E=INT(RND*6)+5:F=INT(RND*6)+5
00600 POKE 62042,E:POKE 62046,F
00610 R=R+1
00620 IF R=20 THEN 640
00630 GOTO 590
00640 F=INT(RND*6)+5
00650 POKE 62046,F
00660 R=R+1
00670 IF R=35 THEN 690
00680 GOTO 640
00690 PLAY 0,4:R=0
00700 G=INT(RND*6)+5:H=INT(RND*6)+5:I=INT(RND*6)+5
00710 POKE 62102,G:POKE 62106,H:POKE 62110,I
00720 R=R+1
00730 IF R=10 THEN 750
00740 GOTO 700
00750 H=INT(RND*6)+5:I=INT(RND*6)+5
00760 POKE 62106,H:POKE 62110,I
00770 R=R+1
00780 IF R=20 THEN 800
00790 GOTO 750
00800 I=INT(RND*6)+5
00810 POKE 62110,I
00820 R=R+1
00830 IF R=35 THEN 850
00840 GOTO 800
00850 PLAY 0,8:IF A=B AND B=C AND A=C OR D=E AND E=F AND D=F OR G=H AND H=I AND
G=I THEN 890
00860 IF A=B AND B=C AND A=C AND D=E AND E=F AND D=F OR A=B AND B=C AND A=C AND
G=H AND H=I AND G=I OR D=E AND E=F AND D=F AND G=H AND H=I AND G=I THEN 910
00870 IF A=B AND B=C AND A=C AND D=E AND E=F AND D=F AND G=H AND H=I AND G=I THE
N 930
00880 PLAY 0,16:CLS:GOTO 230
00890 CLS:CURS 20,9:UNDERLINE:PRINT"CONGRATULATIONS!!":NORMAL
00900 M1=M1+J1:J1=5:GOTO 980
00910 CLS:CURS 1,9:UNDERLINE:PRINT"CONGRATULATIONS! YOU HAVE WON THE DOUBLE-JACK
POT!!!":NORMAL
00920 M1=M1+J1*2:J1=5:GOTO 980
00930 CLS:CURS 1,9:UNDERLINE:PRINT"CONGRATULATIONS! YOU HAVE WON THE TRIPLE-JACK
POT!!!!":NORMAL
00940 M1=M1+J1*3:J1=5:GOTO 980
00950 CLS: CURS 19,9:UNDERLINE:PRINT" YOU ARE NOW BROKE! ":NORMAL
00960 FOR Z=4097 TO 4180:L=USR(15000,Z):NEXT Z:END
00970 CLS:CURS 16,9:UNDERLINE:PRINT" YOU HAVE FINISHED WITH $";M1;"0 ":NORMAL:EN
D
00980 FOR X=4095 TO 3847 STEP -1:L=USR(15000,X):NEXT X
00990 PLAY 0,4:CLS:GOTO 230
```

```
00710 IFA1$(0)=U1$(I)THENLETC=C+1:F1=1:GOTO840
00720 IFF2=1THEN760
00730IFG(2)>1THEN 760
00740 IFI=2THEN760
00750 IFA1$(1)=U1$(I)THENLETC=C+1:F2=1:GOTO840
00760IFF3=1THEN800
00770 IFI=3THEN800
00780 IFG(3)>1THEN800
00790 IFA1$(2)=U1$(I)THENLETC=C+1:F3=1:GOTO840
00800IFF4=1THEN840
00810 IFG(4)>1THEN840
00820 IFI=4 THEN840
00830 IFA1$(3)=U1$(I)THENLETC=C+1:F4=1:GOTO840
00840 NEXTI
00850 F1=0:F2=0:F3=0:F4=0
00860 IFB=4THENGOSUB900
00870 CURS3,S+1:PRINT"b1=";B;" wh=";C
00880 B=0:C=0
00890 GOTO470
00900 Q=0:FORI=0TO3:Q=Q+2:CURS 24+Q,2:PLAY I+1,3;0
,1:PRINTA1$(I);:NEXTI
00910 IFW3=1:W3=2:GOTO1270
00920 IFW3=2:W3=1:GOTO1340
00930 GOTO1410
00940 FORI=1TO16
00950 PRINT[A63 191]
00960 NEXTI
00970 CURS 22,8:PRINT" M A S T E R  M I N D "
00980 PLAY 16;16;20;23;0;20;23,8
00990 PLAY 0,16
01000 CLS
01010 CURS4,3:PRINT"DO YOU WISH INSTRUCTIONS ?";
01020 INPUT"(Y/N)"S3$
01030 IFS3$="Y"ORS3$="y"THEN2000
01040 IFS3$="n"ORS3$="N"THEN1060
01050 GOTO1010
01060 CLS:CURS4,4:PRINT"DO YOU WANT TO PLAY BY YOU
RSELF('1') ,OR AGAINST ANOTHER "\"PERSON('2') ? ";
01070 INPUTG5
01080 IF G5=1THEN180
01090 IFG5=2 THEN1110
01100 GOTO1060
01110 CLS:CURS4,4:INPUT"PRINT THE NAME OF PLAYER O
NE..."Z3$
01120 CURS4,5:INPUT"PRINT THE NAME OF PLAYER TWO..
."Z4$
01130 CURS4,6:PRINT"HOW MANY GAMES DO YOU EACH WIS
H TO PLAY IN THIS CONTEST ";
01140 INPUTV
01150 IFV<1THEN1130
01160 W3=1
01170 GOTO170
01180 PLAY0,20:PRINT\\
01190 IFN<MTHEN1220
01200 PRINTZ3$;" HAS W O N ! ! ! "
01210 GOTO1230
01220 PRINTZ4$;" HAS W O N ! ! ! "
01230 PRINT\\\"DO YOU WISH TO PLAY AGAIN ?(Y/N)";
01240 INPUTS1$
01250 IFS1$="Y"ORS1$="y"THENRUN
01260 END
01270 K=K+1:M=L+M:PLAY0,10:CLS
01280 CURS10,1:PRINTZ3$;" SCORE IS ";M;
01290 CURS10,3:PRINTZ4$;" SCORE IS ";N;
01300 IFK=2*VTHEN1180
01310 PRINT\"IT IS NOW YOUR TURN , ";Z4$
01320 PLAY 0,10
01330 GOTO180
01340 K=K+1:N=L+N:PLAY0,10:CLS
01350 CURS10,1:PRINTZ3$;" SCORE IS ";M;
01360 CURS10,3:PRINTZ4$;" SCORE IS ";N;
01370 IFK=2*VTHEN1180
01380 PRINT\"IT IS NOW YOUR TURN , ";Z3$
01390 PLAY 0,10
01400 GOTO180
01410 PLAY0,6:CLS:T1=T1+1:J1=J1+FLT(L)
01420 PRINT\"YOUR SCORE THAT TURN WAS ";L
01430 PRINT\"YOUR AVERAGE SO FAR IS ";J1/T1
01440 IFFLT(L)>J1/T1:PRINT"YOU HAVE SCORED WORSE T
HAN YOUR ANVERAGE!!"
01450 IFFLT(L)<J1/T1:PRINT"YOU ARE IMPROVING!!!"
01460 PLAY0,50
01470 CLS:PRINT"DO YOU WISH TO PLAY AGAIN ?(Y/N)"
01480 INPUT S2$
01490 IFS2$="Y"ORS2$="y"THEN180
01500 GOTO1010
02000 UNDERLINE:CLS:CURS26,3:PRINT"INSTRUCTIONS":N
ORMAL
02010 PRINT"     THE COMPUTER SECRETLY PUTS 4 COLOR
S BEHIND THE SCREEN IN"
02020 PRINT"ANY ORDER IT WISHES. IT HAS 6 TO SELEC
T FROM (R=RED,B=BLUE"\"G=GREEN,W=WHITE,P=PURPLE,Y=

YELLOW),AND IT CAN DOUBLE COLORS"
02030 PRINT"UP.YOU MUST TRY TO MATCH THE COLOURS I
N THE SAME ORDER AS THE"\"COMPUTER HAS HIDDEN THEM
. CLUES ARE GIVEN AT THE END OF EACH"
02040 PRINT"GUESS(MENTIONED LATER) AND YOU ONLY GE
T TEN CHANCES TO BRAKE"\"THE CODE. WHEN ALL 4 ARE
THE CORRECT COLOURS AND IN THE RIGHT"
02050 PRINT"POSITINS YOU HAVE BROKEN THE COMPUTERS
 HIDDEN CODE, AND IT "\"WILL REVEAL IT FROM BEHIND
 THE SCREEN."
02060 CURS 30,16:PRINT"HIT ANY KEY TO CONTINUE.";
02070R1$=KEY:IFR1$=""THEN2070
02075CLS
02080 UNDERLINE:CURS29,3:PRINT" CLUES ":NORMAL
02090 PRINT\" AFTER EACH TURN, IN THE LEFT HAND CO
LUMN YOU WILL SEE THESE"\"SYMBOLS,'b1' AND 'wh',TH
ESE ARE THE IMPORTANT CLUES"
02100 PRINT\"b1:- IT SHOWS THE NUMBER OF CORRECT C
OLOURS IN THE SAME "\"POSITION AS THE HIDDEN ONES"
02110 PRINT\"wh:- REPRESENTS THE NUMBER OF COLORS
THAT ARE CORRECT BUT IN"\"THE WRONG POSITION"
02120 PRINT\"THE PROBLEM IS DETERMINING WHAT GOES
WHERE BY LOOKING BACK"\"AT THE CLUES FROM ALL YOUR
 TURNS"
02130 CURS30,16:PRINT"HIT ANY KEY TO CONTINUE";
02140 R1$=KEY:IFR1$=""THEN2140
02150 CLS:CURS27,1:UNDERLINE:PRINT"EXAMPLE":NORMAL
02160 PRINT "PLAYERS GUESS - R P G B "
02170 PRINT\"THE ACTUAL CODE(HIDDEN AT THIS STAGE)
"\"                R G B W "
02180 PRINT" THE COMPUTER GIVES THESE CLUES "\"
b1=1 wh=2 "
02190 PRINT" THIS MEANS THE PLAYER HAS 1 CORRECT i
.e 'R'"\"THE PLAYER ALSO HAS TWO CORRECT BUT IN TH
E WRONG POSITIONS"
02200 PRINT"i.e 'G' AND 'B'"\"THE LETTER 'P' IS NO
T THE RIGHT COLOUR AT ALL"
02210 PRINT\\"GOOD LUCK"
02220 CURS30,16:PRINT"HIT ANY KEY TO CONTINUE";
02230 R1$=KEY:IFR1$=""THEN2230
02240 GOTO 1060
```

MICROBEE

KEYWORDS

This program allows words, commands or sequences of up to 15 characters in length to be typed repetitively with just two keystrokes. Twenty six words can be stored by typing TAB then the letter to store it under (ie. A to Z) followed by the characters to be remembered and finally TAB again to mark the end.

To list all the words stored, type LINE FEED twice. (see Table 1.) To recall a word, type LINE FEED then the letter it was stored under. This will

```
ADDR   CODE     LINE   LABEL   MNEM    OPERAND

                00100 ;******** SINLE KEYWORD ENTRY ********
                00110
00C2            00120 VECTOR  EQU     00C2H           ;input vector
8006            00130 GETKEY  EQU     8006H           ;wait for a key into A
800C            00140 DISPLY  EQU     800CH           ;display char in B
00A0            00150 MEMTOP  EQU     00A0H           ;top of memory pointer
A3E9            00160 NORMAL  EQU     0A3E9H          ;normal input driver
                00170
7D00            00180         ORG     7D00H
                00190
7D00 210F7D     00200 INIT    LD      HL,DRIVER
7D03 22C200     00210         LD      (VECTOR),HL     ;Store new input vector
7D06 21FE7C     00220         LD      HL,INIT-2       ;HL=>top of usable mem.
7D09 22A000     00230         LD      (MEMTOP),HL     ;Reset memory size
7D0C C32180     00240         JP      8021H           ;Return to basic
                00250
7D0F CDE9A3     00260 DRIVER  CALL    NORMAL          ;Get a char from keyboard
7D12 C0         00270         RET     NZ              ;Return if none
7D13 FE0A       00280         CP      0AH             ;Check for a LINE FEED
7D15 2806       00290         JR      Z,LINEFD
7D17 FE09       00300         CP      09H             ;Check for a TAB
7D19 2856       00310         JR      Z,TAB
7D1B BF         00320         CP      A               ;Set Z flag
7D1C C9         00330         RET                     ;Return with char in A
                00340
7D1D 21277D     00350 LINEFD  LD      HL,KEYED        ;Reset input vector to
7D20 22C200     00360         LD      (VECTOR),HL     ;    intercept next key
7D23 3EFF       00370 NOKEY   LD      A,0FFH
7D25 B7         00380         OR      A               ;Reset Z flag
7D26 C9         00390         RET                     ;Return as if no key pressed
                00400
7D27 CDE9A3     00410 KEYED   CALL    NORMAL          ;Get char from keyboard
7D2A C0         00420         RET     NZ              ;Return if none
7D2B FE0A       00430         CP      0AH             ;Check for listing
7D2D CAC77D     00440         JP      Z,LIST
7D30 FE60       00450         CP      60H
7D32 3802       00460         JR      C,NOLW1         ;Go if UPPER CASE
7D34 D620       00470         SUB     20H             ;Convert to UPPER CASE
7D36 FE41       00480 NOLW1   CP      'A'
7D38 382D       00490         JR      C,ABORT         ;Go if char not used
7D3A FE5B       00500         CP      'Z'+1
7D3C 3029       00510         JR      NC,ABORT        ;Go if char not used
7D3E D641       00520         SUB     41H             ;A=code for word
7D40 17         00530         RLA                     ;Multiply
7D41 17         00540         RLA                     ;    code by
7D42 17         00550         RLA                     ;    16.
7D43 17         00560         RLA
7D44 C5         00570         PUSH    BC              ;Save BC
7D45 4F         00580         LD      C,A             ;C=LSB of pointer
7D46 0600       00590         LD      B,0
7D48 CB10       00600         RL      B               ;B=MSB of pointer
7D4A 21197E     00610         LD      HL,DATA         ;HL=>first word in table
7D4D 09         00620         ADD     HL,BC           ;HL=>word to bounce back
7D4E C1         00630         POP     BC              ;Restore BC
7D4F 226F7D     00640         LD      (WORD),HL       ;Store word pointer
7D52 215A7D     00650         LD      HL,BOUNCE       ;Reset input vector
7D55 22C200     00660         LD      (VECTOR),HL     ;    to bounce word back.
7D58 18C9       00670         JR      NOKEY           ;Return as if no key pressed
```

greatly increase the speed of typing in a program under BASIC version 5.10.

Listing 1 shows the source code for a 32K Microbee. It can be changed for a 16K Microbee by replacing line 180 with ORG 3DOOH and reassembling. The program is entered from Listing 2 which can be used for 16K or 32K Microbees. Once run this program can be cleared by typing NEW. The keyword will remain enabled even if the Microbee is warm reset.

David Morrison
East Ringwood VIC

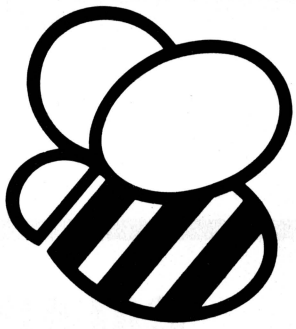
```
ADDR   CODE     LINE   LABEL    MNEM    OPERAND

                00680
7D5A 2A6F7D     00690  BOUNCE   LD      HL,(WORD)       ;HL=>word to bounce
7D5D 7E         00700           LD      A,(HL)          ;A=char to send back
7D5E 23         00710           INC     HL              ;HL=>next char in word
7D5F 226F7D     00720           LD      (WORD),HL       ;Restore word pointer
7D62 B7         00730           OR      A
7D63 2802       00740           JR      Z,ABORT         ;Go if end of word
7D65 BF         00750           CP      A               ;Set Z flag
7D66 C9         00760           RET                     ;Return with char in A
                00770
7D67 210F7D     00780  ABORT    LD      HL,DRIVER       ;Reset input vector
7D6A 22C200     00790           LD      (VECTOR),HL     ;   for normal operation
7D6D 18B4       00800           JR      NOKEY           ;Return as if no
                00810                                           key pressed
7D6F 0000       00820  WORD     DEFW    0000
                00830
7D71 CD0680     00840  TAB      CALL    GETKEY          ;Wait for a key
7D74 FE60       00850           CP      60H
7D76 3802       00860           JR      C,NOLW2         ;Go if UPPER CASE
7D78 D620       00870           SUB     20H             ;Convert to UPPER CASE
7D7A FE41       00880  NOLW2    CP      'A'
7D7C 38E9       00890           JR      C,ABORT         ;Go if char not used
7D7E FE5B       00900           CP      'Z'+1
7D80 30E5       00910           JR      NC,ABORT        ;Go if char not used
7D82 C5         00920           PUSH    BC              ;Save BC
7D83 47         00930           LD      B,A
7D84 CD0C80     00940           CALL    DISPLY          ;Show char to be modified
7D87 D641       00950           SUB     41H             ;A=code for word
7D89 17         00960           RLA                     ;Mupltiply
7D8A 17         00970           RLA                     ;   code by
7D8B 17         00980           RLA                     ;   16.
7D8C 17         00990           RLA
7D8D 4F         01000           LD      C,A             ;C=LSB of pointer
7D8E 0600       01010           LD      B,0
7D90 CB10       01020           RL      B               ;B=MSB of pointer
7D92 21197E     01030           LD      HL,DATA         ;HL=>first word in table
7D95 09         01040           ADD     HL,BC           ;HL=>word to modify
7D96 0E00       01050           LD      C,0             ;Set word length to zero
7D98 063D       01060           LD      B,'='
7D9A CD0C80     01070           CALL    DISPLY          ;Show ready to modify
7D9D 0622       01080           LD      B,'"'
7D9F CD0C80     01090           CALL    DISPLY          ;   the word sign.
7DA2 CD0680     01100  MORE     CALL    GETKEY          ;Wait for key
7DA5 FE09       01110           CP      09              ;Check for TAB
7DA7 2812       01120           JR      Z,EXIT          ;Go if end of word
7DA9 77         01130           LD      (HL),A          ;Store char into word
7DAA 23         01140           INC     HL              ;HL=>next char in word
7DAB FE20       01150           CP      32              ;Check for a control char
7DAD 3002       01160           JR      NC,CONTO        ;Go if not a control char
7DAF 3E5F       01170           LD      A,95            ;_ Signifies control char
7DB1 47         01180  CONTO    LD      B,A
7DB2 CD0C80     01190           CALL    DISPLY          ;Show char of word
7DB5 0C         01200           INC     C               ;Increment word length
7DB6 79         01210           LD      A,C
7DB7 FE0F       01220           CP      15              ;Check for max length
7DB9 20E7       01230           JR      NZ,MORE         ;Get next char in word
7DBB 3600       01240  EXIT     LD      (HL),00         ;Mark end of word
7DBD 0622       01250           LD      B,'"'
```

KEYWORDS

```
ADDR   CODE    LINE   LABEL   MNEM    OPERAND

7DBF CD0C80    01260          CALL    DISPLY          ;Show end of word
7DC2 C1        01270          POP     BC              ;Restore BC
7DC3 3E03      01280 BREAK    LD      A,03
7DC5 BF        01290          CP      A               ;Set Z flag
7DC6 C9        01300          RET                     ;Return as if BREAK pressed
               01310
7DC7 D9        01320 LIST     EXX                     ;Save registers
7DC8 060A      01330          LD      B,0AH           ;B=LINE FEED
7DCA CD0C80    01340          CALL    DISPLY
7DCD 060D      01350          LD      B,0DH           ;B=RETURN
7DCF CD0C80    01360          CALL    DISPLY
7DD2 111A41    01370          LD      DE,411AH        ;D=ascii,E=code of word
7DD5 21197E    01380          LD      HL,DATA         ;HL=>first word in table
7DD8 42        01390 LOOP0    LD      B,D
7DD9 CD0C80    01400          CALL    DISPLY          ;Show ascii char for word
7DDC 063A      01410          LD      B,':'
7DDE CD0C80    01420          CALL    DISPLY
7DE1 0620      01430          LD      B,' '
7DE3 CD0C80    01440          CALL    DISPLY
7DE6 0E1D      01450          LD      C,29            ;C=spacing counter
7DE8 E5        01460          PUSH    HL              ;Save HL
7DE9 7E        01470 LOOP1    LD      A,(HL)          ;Get char from word
7DEA B7        01480          OR      A
7DEB 2810      01490          JR      Z,NEXTWD        ;Go if end of word
7DED FE20      01500          CP      32              ;Check for a control char
7DEF 3002      01510          JR      NC,OK           ;Go if not a control char
7DF1 3E5F      01520          LD      A,95            ;_  Signifies control char
7DF3 47        01530 OK       LD      B,A
7DF4 CD0C80    01540          CALL    DISPLY          ;Show char of word
7DF7 0D        01550          DEC     C               ;Decrement spacing counter
7DF8 280B      01560          JR      Z,NXTWD         ;Go if no more space
7DFA 23        01570          INC     HL              ;HL=>Next char in word
7DFB 18EC      01580          JR      LOOP1           ;Show rest of word
7DFD 0620      01590 NEXTWD   LD      B,' '
7DFF CD0C80    01600          CALL    DISPLY          ;Line up with spaces
7E02 0D        01610          DEC     C
7E03 20F8      01620          JR      NZ,NEXTWD
7E05 14        01630 NXTWD    INC     D               ;Next ascii for word
7E06 E1        01640          POP     HL              ;Restore  HL
7E07 C5        01650          PUSH    BC              ;Save BC
7E08 011000    01660          LD      BC,10H
7E0B 09        01670          ADD     HL,BC           ;HL=>next word
7E0C C1        01680          POP     BC              ;Restore BC
7E0D 1D        01690          DEC     E               ;Next code for word
7E0E 20C8      01700          JR      NZ,LOOP0        ;Go until no more words
7E10 D9        01710          EXX                     ;Restore registers
7E11 210F7D    01720          LD      HL,DRIVER       ;Restore input driver
7E14 22C200    01730          LD      (VECTOR),HL     ;   for normal operation
7E17 18AA      01740          JR      BREAK           ;Return as if BREAK pressed
               01750
               01760
01A0           01770 DATA     DEFS    1A0H
7FB9           01780 HIMEM    EQU     $
               01790
0000           01800          END
00000 Total errors

HIMEM   7FB9    NXTWD   7E05    OK      7DF3    NEXTWD  7DFD
LOOP1   7DE9    LOOP0   7DD8    BREAK   7DC3    CONTO   7DB1
EXIT    7DBB    MORE    7DA2    NOLW2   7D7A    BOUNCE  7D5A
WORD    7D6F    DATA    7E19    ABORT   7D67    NOLW1   7D36
LIST    7DC7    NOKEY   7D23    KEYED   7D27    TAB     7D71
LINEFD  7D1D    DRIVER  7D0F    INIT    7D00    NORMAL  A3E9
MEMTOP  00A0    DISPLY  800C    GETKEY  8006    VECTOR  00C2
```

Table 1.

Words stored under keyboard characters



A: AUTO	B: GOSUB
C: CURS	D: EDIT_
E: EDIT	F: FOR
G: GOTO	H: HIRES
I: INPUT	J: EDASM_
K: PEEK(	L: LIST_
M: RENUM	N: NORMAL
O: POKE	P: PRINT
Q: PLOT	R: RUN_
S: SET(	T: THEN
U: UNDERLINE	V: RETURN
W: RESET(	X: NEXT
Y: PLAY	Z: ZONE

Listing 2.

```
00100 POKE 32000,128
00110 IF PEEK(32000)=128 THEN LET Z=32000 ELSE LET Z=15616
00120 PRINT"Single keyword entry."
00130 PRINT"Storing program into ";
00140 IF Z=32000 THEN PRINT"32K"; ELSE PRINT"16K";
00150 PRINT" memory."
00160 FOR A=Z TO Z+687:READ B:POKE A,B:NEXT A
00170 IF Z=32000 THEN 180
00171 POKE Z+2,61:POKE Z+8,60:POKE Z+31,61:POKE Z+47,61
00172 POKE Z+76,62:POKE Z+81,61:POKE Z+84,61:POKE Z+92,61
00173 POKE Z+97,61:POKE Z+105,61:POKE Z+148,62:POKE Z+215,62
00174 POKE Z+275,61
00180 X=USR(Z)
10010 DATA 33,15,125,34,194,0,33,254,124,34,160,0,195,33,128
10020 DATA 205,233,163,192,254,10,40,6,254,9,40,86,191,201,33
10030 DATA 39,125,34,194,0,62,255,183,201,205,233,163,192,254
10040 DATA 10,202,199,125,254,96,56,2,214,32,254,65,56,45,254
10050 DATA 91,48,41,214,65,23,23,23,23,197,79,6,0,203,16,33
10060 DATA 25,126,9,193,34,111,125,33,90,125,34,194,0,24,201
10070 DATA 42,111,125,126,35,34,111,125,183,40,2,191,201,33
10080 DATA 15,125,34,194,0,24,180,99,127,205,6,128,254,96,56
10090 DATA 2,214,32,254,65,56,233,254,91,48,229,197,71,205,12
10100 DATA 128,214,65,23,23,23,23,79,6,0,203,16,33,25,126,9
10110 DATA 14,0,6,61,205,12,128,6,34,205,12,128,205,6,128,254
10120 DATA 9,40,18,119,35,254,32,48,2,62,95,71,205,12,128,12
10130 DATA 121,254,15,32,231,54,0,6,34,205,12,128,193,62,3,191
10140 DATA 201,217,6,10,205,12,128,6,13,205,12,128,17,26,65
10150 DATA 33,25,126,66,205,12,128,6,58,205,12,128,6,32,205
10160 DATA 12,128,14,29,229,126,183,40,16,254,32,48,2,62,95
10170 DATA 71,205,12,128,13,40,11,35,24,236,6,32,205,12,128
10180 DATA 13,32,248,20,225,197,1,16,0,9,193,29,32,200,217,33
10190 DATA 15,125,34,194,0,24,170,65,85,84,79,0,0,0,0,0,0,0
10200 DATA 0,0,0,0,0,71,79,83,85,66,32,0,0,0,0,0,0,0,0,0,67
10210 DATA 85,82,83,32,0,0,0,0,0,0,0,0,0,0,69,68,73,84,13
10220 DATA 0,0,0,0,0,0,0,0,0,0,0,69,68,73,84,32,0,0,0,0,0,0
10230 DATA 0,0,0,0,0,70,79,82,32,0,0,0,0,0,0,0,0,0,0,0,71
10240 DATA 79,84,79,32,0,0,0,0,0,0,0,0,0,0,72,73,82,69,83
10250 DATA 0,0,0,0,0,0,0,0,0,0,0,73,78,80,85,84,0,0,0,0,0,0
10260 DATA 0,0,0,0,0,0,69,68,65,83,77,13,0,0,0,0,0,0,0,0,0,0,80
10270 DATA 69,69,75,40,0,0,0,0,0,0,0,0,0,0,76,73,83,84,13
10280 DATA 0,0,0,0,0,0,0,0,0,0,0,0,82,69,78,85,77,32,0,0,0,0,0
10290 DATA 0,0,0,0,0,78,79,82,77,65,76,0,0,0,0,0,0,0,0,0,0,80
10300 DATA 79,75,69,32,0,0,0,0,0,0,0,0,0,0,80,82,73,78,84
10310 DATA 32,0,0,0,0,0,0,0,0,0,0,80,76,79,84,32,0,0,0,0,0,0
10320 DATA 0,0,0,0,0,82,85,78,13,0,0,0,0,0,0,0,0,0,0,0,83
10330 DATA 69,84,40,0,0,0,0,0,0,0,0,0,0,0,32,84,72,69,78,32
10340 DATA 0,0,0,0,0,0,0,0,0,0,85,78,68,69,82,76,73,78,69,0
10350 DATA 0,0,0,0,0,0,82,69,84,85,82,78,0,0,0,0,0,0,0,0,0,0
10360 DATA 82,69,83,69,84,40,0,0,0,0,0,0,0,0,78,69,88,84
10370 DATA 32,0,0,0,0,0,0,0,0,0,0,80,76,65,89,32,0,0,0,0,0
10380 DATA 0,0,0,0,0,0,90,79,78,69,32,0,0
```

GALACTIC FIGHTER

Well, here's a game for the Microbee and yes, it is another variation on the old Star Trek game. The only difference is that I wrote it.

Instructions

Just run the program. Once out of the menu, the game will begin. The commands are T for thrust, S for shields, F for torpedoes, D for display, C for computer, W for game save and X for self-destruct.

In the Thrust mode there are two choices, T for controlled thrust and N for hyperspace.

In the shield routine, T for transfer and S for shield status change.

Within shield change, F is forward, R is rear, P is port, S is starboard, U is shield up and D is shield down.

Cursor control for Computer direction/distance calculator 1 is W=up, Z=down, A=left and S=right. A RETURN will exit cursor control and print co-ordinates of cursor position, direction of cursor from Viper and the distance. Another RETURN is then awaited before the command mode is entered.

There you have it. Any mistakes made will be prompted by an error message. It may be useful to place a CTRL G character at the end of every error message when typing the program in. This will get the player's attention if an error is made.

I have noticed that the energy for the Viper is usually insufficient. Try replacing A3=RND* FLT(E)*250+20000 in line 350 by A#=RND*FLT(E)* 400+ 20000. Have fun!

Jon Barnett
Northmead NSW

```
00100 DATA 33,210,0,195,162,148
00110 POKE 162,80:POKE 163,1:RESTORE 100:FOR A=336 TO 341:READ B:POKE A,B:NEXT A:POKE 140,1:POKE 220
,20
00120 GOTO 210
00130 X=INT(RND*100)+1:Y=INT(RND*100)+1:A2$=A0$(Y):IF A2$(X,X)<>"$" THEN 130 ELSE LET A2$=A2$(1,X-
1)+A1$+A2$(X+1):A0$(Y)=A2$:RETURN
00140 CLS:CURS 1,1:FOR C=1 TO 13:PRINT CA23 165]:NEXT C:PCG:D=B+12:FOR C=B+10 TO B STEP-1:CURS 2,D-C
:A1$=A0$(C):PRINT A1$(A,A+20):NEXT C:NORMAL
00150 CURS 30,3:PRINT"Condition        "S0$:CURS 30,4:PRINT"Co-ordinates    "CI4 X]","CI4 Y]:CURS 30
,5:PRINT"Energy          "CF8.1 A3]:CURS 30,6:PRINT"Cylons Left     "CI3 E]
00160 CURS 30,7:PCG:PRINT"Shields:";:NORMAL:PRINT SPC(8)CF6.1 B3]:CURS 30,8:PRINT"Forward" SPC(10)A4
$(0):CURS 30,9:PRINT"Rear" SPC(13)A4$(1):CURS 30,10:PRINT"Starboard" SPC(8) A4$(2)
00170 CURS 30,11:PRINT"Port" SPC(13)A4$(3):RETURN
00180 CURS 30,14:PRINT"Insufficient energy available.":CURS 30,15:PRINT"Returning to Command mode.":
FOR C=1 TO 300:NEXT C:GOTO 370
00190 DATA 24,24,24,24,153,153,153,165,102,231,219,219,219,219,255,0,0,24,60,126,126,255,255,231
,102,165,153,129,0,0,0,0,0,0,0,8,73,42,28,42,73,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0
00200 DATA 0,0,0,0,0,0,0,60,60,60,60,60,60,60,60,60,60,60,60,0,0,0,0,0,0,16,16,16,16,16,16,40,108,16
,16,0
00210 IN#0:OUT#0:INVERSE:NORMAL:RESTORE 190:A=64016:FOR A=A TO A+95:READ B:POKE A,B:NEXT A
00220 DATA 62,32,2,62,11,17,0,245,33,65,240,229,1,21,0,237,176,225,213,17,64,0,25,209,61,32,240,33,6
5,240,14,11,62,160,17,64,0,6,21,229,119,35,16,252,225,25,13,32,244,201
00230 DATA 33,0,245,17,65,240,62,11,213,1,21,0,237,176,209,229,33,64,0,25,235,225,61,32,239,201
00240 RESTORE 220:A=520:FOR A=A TO A+75:READ B:POKE A,B:NEXT A
00250 CLS:CURS 23,1:PRINT CA18 35]\TAB(23)"#GALACTIC FIGHTER#"\TAB(23)CA18 35]
00260 A1$="                ****Prosram by J.L.Barnett                ****Press any key to contin
```

```
ue                    ":SPEED 10
00270 FOR A=1 TO 109:CURS 23,7:PRINT A1$(;A,A+17):IF KEY="" THEN NEXT A:GOTO 270 ELSE NEXT*A 280
00280 CURS 23,7:PRINT"**By J.L.Barnett**"
00290 A1$="           1. New Game           2. Old Game           3. Saved Game           ":CURS 23,13:P
RINT"Option:"
00300 SPEED 20:FOR A=1 TO 75:CURS 30,13:PRINT A1$(;A,A+9):A2$=KEY:IF A2$="" THEN NEXT A :GOTO 300 EL
SE NEXT*A 310
00310 SPEED 0:IF A2$="1" THEN CLEAR:CURS 30,13:PRINT"New Game   ":GOTO 320 ELSE IF A2$="2" AND Z=1 T
HEN CURS 30,13:PRINT"Old Game   ":GOTO 370 ELSE IF A2$="3" THEN 1530 ELSE 300
00320 STRS(11000):E=INT(RND*20)+30:S=INT(RND*30)+40:DIM T(S,1),U(E,1),A0(100),A4(3)
00330 Z=1:PRMT(:):FOR A=1 TO 100:A0$(A)="$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$":NEXT A
00340 A1$=CHR(34):FOR A=1 TO E:GOSUB 130:NEXT A:A1$="#":FOR A=1 TO S:GOSUB 130:NEXT A:A1$="!":GOSUB
130
00350 A3=RND*FLT(E)*250+20000:B3=500:A3=A3-500:FOR A=0 TO 3:A4$(A)="Down":NEXT A
00360 FN0=#*.01745329
00370 S0$="  GREEN":A=X-10:B=Y-5:IF A<1 THEN LET A=1 ELSE IF A>80 THEN LET A=80
00380 IF B<1 THEN LET B=1 ELSE IF B>90 THEN LET B=90
00390 A2$=CHR(34):FOR D=B TO B+10:A1$=A0$(D):A1$=A1$(;A,A+20):FOR C=1 TO 20:F=SEARCH(A1$,A2$,C):IF F
=0 THEN NEXT*C 440 ELSE IF RND<.6 THEN 430
00400 F=F+A-1:A5=SGN(FLT(F-X)):A6=SGN(FLT(D-Y)):IF D-Y=0 THEN LET A7=58*A5 ELSE LET A7=FLT(F-X)/FLT(
D-Y)
00410 B0=150:IF ABS(A7)<=1 AND A6=1 THEN LET G=0 ELSE IF ABS(A7)<=1 AND A6=-1 THEN LET G=1 ELSE IF A
5=1 THEN LET G=2 ELSE LET G=3
00420 IF A4$(G)="Down" THEN LET A3=A3-B0*(.9-RND*.15) ELSE LET B1=RND*.1+.1:B3=B3-B0*B1:A3=A3-(B0-B0
*B1)*(.9-RND*.15)
00430 NEXT C
00440 NEXT D
00450 FOR D=B TO B+10:A1$=A0$(D):A1$=A1$(;A,A+20):F=SEARCH(A1$,A2$):IF F=0 THEN 500 ELSE IF RND<.5 T
HEN 500 ELSE LET F=F+A-1
00460 IF RND<.5 THEN LET G=F+1 ELSE LET G=F-1
00470 IF RND<.5 THEN LET H=D+1 ELSE LET H=D-1
00480 IF H<1 OR G<1 OR H>100 OR G>100 THEN 500
00490 A1$=A0$(H):IF A1$(;G,G)<>"$" THEN 500 ELSE LET A1$=A1$(;1,G-1)+A2$+A1$(;G+1):A0$(H)=A1$:A1$=A0
$(D):A1$=A1$(;1,F-1)+"$"+A1$(;F+1):A0$(D)=A1$
00500 NEXT D
00510 C=INT(B3)/100:F=0:FOR D=0 TO 3:IF A4$(D)="Up" THEN LET F=F+1
00520 NEXT D:IF C>=F OR F=0 THEN 530 ELSE LET G=INT(RND*4):A4$(G)="Down"
00530 A2$=CHR(34):FOR C=B TO B+10:A1$=A0$(C):A1$=A1$(;A,A+20):IF SEARCH(A1$,A2$)<>0 THEN LET S0$="**
YELLOW**":NEXT*C 540 ELSE NEXT C:GOTO 570
00540 FOR D=Y-3 TO Y+3:IF D<1 OR D>100 THEN NEXT D:GOTO 570 ELSE LET A1$=A0$(D):FOR C=X-3 TO X+3:IF
A1$(;C,C)=CHR(34) THEN LET S0$="***RED***":NEXT*C 560 ELSE NEXT C:NEXT D
00550 GOTO 570
00560 NEXT*D 570
00570 D=0:GOSUB 140:IF A3<10 THEN 1380 ELSE CURS 30,13:PCG:PRINT"COMMAND:";:NORMAL
00580 D=D+1:A1$=KEY:IF D>200 THEN 370 ELSE IF A1$="T" OR A1$="t" THEN PRINT"Thrust":GOTO 660
00590 IF A1$="S" OR A1$="s" THEN PRINT"Shields":GOTO 790
00600 IF A1$="F" OR A1$="f" THEN PRINT"Torpedoes":GOTO 980
00610 IF A1$="D" OR A1$="d" THEN PRINT"Display":GOTO 1080
00620 IF A1$="C" OR A1$="c" THEN PRINT"Computer":GOTO 1160
00630 IF A1$="W" OR A1$="w" THEN PRINT"Save":CURS 1,14:PRINT [A127 32]:GOTO 1470
00640 IF A1$="X" OR A1$="x" THEN CLEAR:GOTO 1380
00650 IF A1$="" THEN 580 ELSE CURS 30,14:PRINT"Computer unable to interpret.":CURS 30,15:PRINT"Pleas
e re-enter.":CURS 38,13:GOTO 580
00660 IF A3<100 THEN 180 ELSE CURS 1,14:PRINT [A127 32]:CURS 1,14:PCG:PRINT"Mode:":NORMAL:CURS 1,15:
PRINT"Thrusters or Hyperspace";
00670 A1$=KEY:IF A1$="T" OR A1$="t" THEN 720 ELSE IF A1$="H" OR A1$="h" THEN LET D=USR(523):GOTO 690
 ELSE IF A1$="" THEN 670
00680 CURS 30,14:PRINT"Computer unable to translate.":CURS 30,15:PRINT"Please re-enter.":CURS 6,14:G
OTO 670
00690 IF A3<100 THEN 180 ELSE LET A3=A3-100
00700 CURS 1,14:PRINT [A127 32]:CURS 1,14:PRINT"Hyperspace":G=INT(RND*100)+1:H=INT(RND*100)+1:A1$=A0
$(H):IF A1$(;G,G)<>"$" THEN 1390 ELSE LET A1$=A1$(;1,G-1)+"!"+A1$(;G+1):A0$(H)=A1$
00710 A1$=A0$(Y):A1$=A1$(;1,X-1)+"$"+A1$(;X+1):A0$(Y)=A1$:X=G:Y=H:GOTO 370
00720 CURS 1,14:PRINT [A127 32]:CURS 1,14:INPUT"Direction"A5:A5=FN0(A5):A6=SIN(A5):A7=COS(A5)
00730 CURS 1,15:INPUT"Distance"C:C=INT(ABS(FLT(C))):IF C=0 OR C>10 THEN CURS 30,14:PRINT"Computer de
tects incorrect data.":CURS 1,15:PRINT [A29 32]"Re-enter please.":GOTO 730
00740 IF A3<FLT(C)*15*(ABS(A6)+ABS(A7)+1) THEN 180 ELSE LET B0=FLT(X):B1=FLT(Y):G=X:H=Y:FOR C=1 TO C
:B0=B0+A6:B1=B1+A7:IF B1>100 THEN LET B1=102-B1 ELSE IF B1<1 THEN LET B1=100-B1
00750 IF B0>100 THEN LET B0=102-B0 ELSE IF B0<1 THEN LET B0=100-B0
00760 D=INT(B0):A1$=A0$(INT(B1)):IF A1$(;D,D)<>"$" AND A1$(;D,D)<>"!" THEN NEXT*C 780 ELSE LET G=INT
(B0):H=INT(B1):NEXT C
```

```
00770 A3=A3-FLT(C)*15*(ABS(A6)+ABS(A7)+1):A1$=A0$(Y):A1$=A1$(;1,X-1)+"$"+A1$(;X+1):A0$(Y)=A1$:X=G:Y=
H:A1$=A0$(Y):A1$=A1$(;1,X-1)+"!"+A1$(;X+1):A0$(Y)=A1$:GOTO 370
00780 CURS 30,14:PRINT"Computer shut ensines down due to":CURS 30,15:PRINT"an obstacle in path. ":FO
R D=0 TO 300:NEXT D:GOTO 770
00790 CURS 1,14:PRINT [A127 32]:CURS 1,14:PRINT"Transfer or Shield Change";
00800 A1$=KEY:IF A1$="T" OR A1$="t" THEN CURS 1,14:PRINT [A127 32]:GOTO 930 ELSE IF A1$="S" OR A1$="
s" THEN CURS 1,14:PRINT [A127 32]:GOTO 820
00810 IF A1$="" THEN 800 ELSE CURS 30,14:PRINT"Computer unable to decode data.":CURS 30,15:PRINT"Re-
enter please.":GOTO 800
00820 IF A3<15 THEN 180 ELSE CURS 1,14:PRINT"Shield:";:C=4
00830 A1$=KEY:IF A1$="F" OR A1$="f" THEN PRINT"Forward":C=0 ELSE IF A1$="R" OR A1$="r" THEN PRINT"Re
ar":C=1 ELSE IF A1$="S" OR A1$="s" THEN PRINT"Starboard":C=2
00840 IF A1$="P" OR A1$="p" THEN PRINT"Port":C=3 ELSE IF A1$="" THEN 830
00850 IF C<>4 THEN 870
00860 CURS 30,14:PRINT"Computer detects incorrect data.":CURS 30,15:PRINT"Please re-enter.":CURS 8,1
4:GOTO 830
00870 F=0:A1$="Up":FOR D=0 TO 3:A2$=A4$(D):F=F+SEARCH(A2$,A1$):NEXT D:CURS 30,14:PRINT [A96 32]:CURS
 1,15:PRINT"Status:";
00880 A1$=KEY:IF A1$="U" OR A1$="u" THEN LET A1$="Up":GOTO 920 ELSE IF A1$="D" OR A1$="d" THEN LET A
1$="Down":GOTO 900
00890 IF A1$="" THEN 880 ELSE CURS 30,14:PRINT"Computer unable to translate.":CURS 30,15:PRINT"Pleas
e re-enter.":CURS 8,15:GOTO 880
00900 IF A1$=A4$(C) THEN CURS 30,14:PRINT"Shield already at status.    ":CURS 30,15:PRINT"Returning
to Command Mode.":FOR C=1 TO 300:NEXT C:GOTO 370
00910 PRINT A1$:CURS 47,8+C:PRINT A1$ "    ":A4$(C)=A1$:A3=A3-15:FOR C=1 TO 300:GOTO 370
00920 IF INT(B3/100)<F+1 THEN CURS 30,14:PRINT"Unable to raise another shield.":CURS 30,15:PRINT"Ret
urning to Shield mode.":FOR C=1 TO 250:NEXT C:GOTO 790 ELSE GOTO 900
00930 CURS 1,14:PCG:PRINT"Transfer Energy:":NORMAL:CURS 1,15:PRINT"Shields available:";
00940 A1$=KEY:IF A1$="" THEN 940
00950 C=INT(VAL(A1$)):IF C<1 OR C>4 THEN CURS 30,14:PRINT"Data input is scrambled.":CURS 30,15:PRINT
"Re-enter please.":GOTO 940 ELSE IF A3+B3<FLT(C)*100+20 THEN 180
00960 CURS 19,15:PRINT A1$:IF C<=INT(B3)/100 THEN CURS 30,14:PRINT"Shields already available.  ":CUR
S 30,15:PRINT"Returning to Shield mode.":FOR C=0 TO 500:NEXT C:GOTO 790
00970 A3=A3+B3-FLT(C)*100-20:B3=FLT(C)*100:GOTO 370
00980 IF A3<30 THEN 180 ELSE CURS 1,14:PRINT [A127 32]:IF A3<30 THEN 180 ELSE CURS 1,14:INPUT"Direct
ion"A5:A5=FNO(A5):A6=SIN(A5):A7=COS(A5):A3=A3-30
00990 CURS 1,15:INPUT"Distance"C:C=INT(ABS(FLT(C))):IF C=0 OR C>5 THEN CURS 30,14:PRINT"Computer det
ects incorrect data.":CURS 1,15:PRINT [A29 32]"Re-enter please.":GOTO 990
01000 B0=FLT(X):B1=FLT(Y):G=X:H=Y:D=1:FOR C=1 TO C:B0=B0+A6:B1=B1+A7:IF INT(B1)>100 OR INT(B0)>100 O
R INT(B1)<1 OR INT(B0)<1 THEN NEXT*C 1030 ELSE IF D<>1 THEN POKE D,32
01010 D=INT(B0):A1$=A0$(INT(B1)):IF A1$(;D,D)=CHR$(34) THEN NEXT*C 1040 ELSE IF A1$(;D,D)="*" THEN NE
XT*C 1070
01020 IF INT(B0)=X AND INT(B1)=Y THEN NEXT C ELSE LET C=D-A+1:H=11+B-INT(B1):D=61440+64*H+G:POKE D,1
66:FOR F=1 TO 100:NEXT F:NEXT C
01030 CURS 30,14:PRINT"Torpedo exploded harmlessly at   ":CURS 30,15:PRINT INT(B0)","INT(B1)"" SPC(1
2):FOR C=1 TO 300:NEXT C:GOTO 370
01040 C=INT(RND*100)+1:IF C<=25 THEN CURS 30,14:PRINT"Torpedo ineffective as the       ":CURS 30,15:
PRINT"Cylon Ship had shields up.":FOR C=1 TO 300:NEXT C:GOTO 370
01050 A1$=A1$(;1,D-1)+"$"+A1$(;D+1):A0$(INT(B1))=A1$:D=D+61441+64*(11+B-INT(B1))-A:D=USR(520,D):FOR
G=1 TO 10:D=USR(570):FOR H=1 TO 100:NEXT H:D=USR(523):FOR H=1 TO 100:NEXT H:NEXT G
01060 D=USR(570):CURS 30,14:PRINT"Cylon Attack Craft destroyed.   ":CURS 30,15:PRINT [A33 32]:PLAY 1
5;15;15;15;15;15;15;E=E-1:FOR C=1 TO 300:NEXT C:IF E=0 THEN 1420 ELSE GOTO 370
01070 CURS 30,14:PRINT"Star absorbed the torpedo at    ":CURS 30,15:PRINT INT(B0)","INT(B1)"" SPC(12
):FOR C=1 TO 300:NEXT C:GOTO 370
01080 IF A3<10 THEN 180 ELSE CURS 1,14:PRINT [A127 32]:A3=A3-10
01090 CURS 1,14:PRINT [A29 32]
01100 CURS 1,14:INPUT"Co-ordinates"A,B:IF A<1 OR B<1 OR A>100 OR B>100 THEN CURS 30,14:PRINT"Compute
r detects incorrect data.":CURS 30,15:PRINT"Please re-enter.":GOTO 1090
01110 G=A:H=B:A=A-10:B=B-5:IF A>80 THEN LET A=80 ELSE IF A<1 THEN LET A=1
01120 IF B>90 THEN LET B=90 ELSE IF B<1 THEN LET B=1
01130 CLS:FOR C=1 TO 13:CURS 20,C:PRINT [A23 165]:NEXT C:PCG:D=B+12:FOR C=B+10 TO B STEP-1:CURS 21,D
-C:A1$=A0$(C):PRINT A1$(;A,A+20):NEXT C:NORMAL
01140 D=61460+64*(11+B-H)+G-A:POKE D,216:A1$="Display:"+STR(G)+","+STR(H):C=(23-LEN(A1$))/2:CURS 20+
C,14:PRINT A1$
01150 IF KEY="" THEN 1150 ELSE GOTO 370
01160 CURS 1,14:PRINT [A127 32]:CURS 1,14:PRINT"Mode:";
01170 A1$=KEY:IF A1$="" THEN 1170 ELSE LET C=INT(VAL(A1$)):IF C<1 OR C>2 THEN CURS 30,14:PRINT"Compu
ter unable to decode data.":CURS 30,15:PRINT"Re-enter please.":CURS 6,14:GOTO 1170
01180 IF C=2 THEN 1320 ELSE IF A3<15 THEN 180 ELSE CURS 1,14:PRINT [A127 32]:CURS 1,14:PRINT"Directi
on/Distance 1":C=A:D=B:G=61441:H=G+64*(11+B-D)+C-A:F=PEEK(H):POKE H,216
01190 FOR I=1 TO 100:A1$=KEY:IF A1$<>"" THEN NEXT*I 1200 ELSE NEXT I:POKE H,F:FOR I=1 TO 100:A1$=KEY
:IF A1$<>"" THEN NEXT*I 1200 ELSE NEXT I:POKE H,216:GOTO 1190
```

```
01200 POKE H,216:IF A1$=CHR(13) OR A1$=CHR(13) THEN 1230 ELSE IF A1$="W" OR A1$="w" AND D<B+10 THEN
LET D=D+1 ELSE IF A1$="Z" OR A1$="z" AND D>B THEN LET D=D-1
01210 IF A1$="S" OR A1$="s" AND C<A+20 THEN LET C=C+1 ELSE IF A1$="A" OR A1$="a" AND C>A THEN LET C=
C-1 ELSE IF A1$<>"W" AND A1$<>"w" AND A1$<>"Z" AND A1$<>"z" THEN 1190
01220 POKE H,F:H=G+64*(11+B-D)+C-A:F=PEEK(H):POKE H,216:GOTO 1190
01230 A3=A3-15:IF C=X AND D=Y THEN CURS 30,14:PRINT"Computer replies that the":CURS 30,15:PRINT"posi
tion supplied is ship's.":FOR C=1 TO 300:NEXT C:GOTO 370
01240 A5=FLT(C-X):A6=FLT(D-Y):B4=SQR(A5*A5+A6*A6):IF A6=0 THEN LET A7=90:GOTO 1280 ELSE LET A7=A5/A6
:IF A7=0 THEN 1280 ELSE IF ABS(A7)>=.1 THEN 1270
01250 IF A7<=.0174551 THEN LET A7=1 ELSE IF A7<=.0349207 THEN LET A7=2 ELSE IF A7<=.0524077 THEN LET
 A7=3 ELSE IF A7<=.0699268 THEN LET A7=4 ELSE IF A7<=.0874887 THEN LET A7=5
01260 IF A7>0 THEN 1280 ELSE LET A7=6:GOTO 1280
01270 A7=ATAN(ABS(A7)):A7=A7*57.295779
01280 A5=SGN(A5):A6=SGN(A6):IF A5=1 AND A6=-1 THEN LET A7=180-A7 ELSE IF A5=-1 AND A6=-1 THEN LET A7
=A7+180 ELSE IF A5=-1 AND A6=1 THEN LET A7=360-A7
01290 IF A5=0 AND A6=-1 THEN LET A7=180 ELSE IF A5=-1 AND A6=0 THEN LET A7=A7+180
01300 CURS 30,14:PRINT"Distance to"C","D" is"[F6.1 B4]:CURS 30,15:PRINT"Direction is"[F6.1 A7]
01310 IF KEY="" THEN 1310 ELSE 370

01320 IF A3<20 THEN 180 ELSE CURS 1,14:PRINT [A127 32]:CURS 1,14:PRINT"Direction/Distance 2"
01330 CURS 1,15:PRINT [A29 32]
01340 CURS 1,15:INPUT"Co-ordinates"C,D:IF C<1 OR D<1 OR C>100 OR D>100 THEN CURS 30,14:PRINT"Compute
r detects incorrect data.":CURS 30,15:PRINT"Please re-enter.":GOTO 1330
01350 IF ABS(FLT(C-X))>10 OR ABS(FLT(D-Y))>5 THEN GOTO 1370 ELSE CURS 30,14:PRINT"Transferring to Di
stance/          ":CURS 30,15:PRINT"Direction Mode 1":H=61441+64*(11+B-D)+C-A
01360 FOR F=1 TO 500:NEXT F:POKE H,216:GOTO 1230
01370 CURS 30,14:PRINT [A33 32]:CURS 30,15:PRINT [A33 32]:A3=A3-20:GOTO 1240
01380 A=USR(523)
01390 FOR G=1 TO 10:D=USR(570):FOR H=1 TO 100:NEXT H:D=USR(523):FOR H=1 TO 100:NEXT H:NEXT G:CLS:PRI
NT"       The engines of your Colonial Viper have exploded.You save"
01400 PRINT"your life for the safety of the Colonial fleet.You will be       remembered by all for yo
ur sacrifice.A minute's silence will be held in your memory."
01410 FOR A=1 TO 1000:NEXT A:GOTO 1440
01420 CLS:PRINT"       You have cleared the galaxy of all Cylon ships.The Council of Twelve has award
ed you the Star of Cobald.At a special       ceremony,Adama himself,will pin ";
01430 PRINT"the medal on you.You are a       Colonial Warrior and an officer of the Battlestar Galact
ica.    Duty and honour comes first."
01440 CURS 1,8:PRINT"The Cylon menace has not yet been stopped.Their ships continue  to reach furthe
r into the depths of space.The Cylons must be    delayed,so the Battlestar";
01450 PRINT" and the accompanying ships can        increase the distance between themselves and the
enemy."\\"Are there any Warriors willing to delay the Cylon attack force?"
01460 A1$=KEY:IF A1$="Y" OR A1$="y" THEN CLEAR:GOTO 250 ELSE IF A1$="N" OR A1$="n" THEN CLS:FOR A=23
04 TO 8192:POKE A,0:NEXT A:FOR A=128 TO 500:POKE A,0:NEXT A:NEW ELSE 1460
01470 CURS 1,14:PRINT"Please wait.":A2$=CHR(34):G=0:FOR D=1 TO 100:A1$=A0$(D):FOR C=1 TO 50:F=SEARCH
(A1$,A2$,C):IF F=0 THEN NEXT*C 1480 ELSE LET U(G,0)=F:U(G,1)=D:G=G+1:NEXT C
01480 NEXT D:H=0:FOR D=1 TO 100:A1$=A0$(D):FOR C=1 TO 70:F=SEARCH(A1$,"#",C):IF F=0 THEN NEXT*C 1490
 ELSE LET T(H,0)=F:T(H,1)=D:H=H+1:NEXT C
01490 NEXT D:H=H-1:G=G-1:CURS 1,15:PRINT"Start recording.":FOR D=1 TO 1500:NEXT D:CURS 1,15:PRINT"No
w SAVEing      ":OUT#2:FOR A=1 TO 3:PRINT [A33 33]:NEXT A:PRINT"$$$$$    GAMEVAR"
01500 PRINT [I3 1]","G","H","E","X","Y","Z","A3","B3:FOR A=0 TO 3:PRINT A4$(A):NEXT A
01510 PRINT [I3 2]:FOR A=0 TO H:PRINT T(A,0):NEXT A:PRINT [I3 3]:FOR A=0 TO H:PRINT T(A,1):NEXT A:PR
INT[I3 4]:FOR A=0 TO G:PRINT U(A,0):NEXT A
01520 PRINT [I3 5]:FOR A=0 TO G:PRINT U(A,1):NEXT A:PRINT [A5 63][A5 26]:OUT#0:GOTO 370
01530 CURS 30,13:PRINT"Saved Game":CLEAR:DIM A0(100),A4(3):IN#2:OUT#0:OUT#0 OFF
01540 INPUT A1$:IF A1$(;1,5)<>"$$$$$" AND A1$(;10)<>"GAMEVAR" THEN 1540 ELSE OUT#0:CURS 30,13:PRINT"
Loading    ":OUT#0 OFF
01550 INPUT A1$,G,H,E,X,Y,Z,A3,B3:FOR A=0 TO 3:INPUT A4$(A):NEXT A:IF VAL(A1$) THEN 1640 ELSE DIM U(
G,1),T(H,1)
01560 INPUT A1$:IF VAL(A1$)<>2 THEN 1640 ELSE FOR A=0 TO H:INPUT T(A,0):NEXT A:INPUT A1$:IF VAL(A1$)
<>3 THEN 1640 ELSE FOR A=0 TO H:INPUT T(A,1):NEXT A
01570 INPUT A1$:IF VAL(A1$)<>4 THEN 1640 ELSE FOR A=0 TO G:INPUT U(A,0):NEXT A:INPUT A1$:IF VAL(A1$)
<>5 THEN 1640 ELSE FOR A=0 TO G:INPUT U(A,1):NEXT A
01580 OUT#0:IN#0:CURS 23,15:PRINT"Loaded"
01590 STRS(11000):PRMT(:):FOR A=1 TO 100:A0$(A)="$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$":NEXT A
01600 FOR A=0 TO H:B=T(A,1):C=T(A,0):A1$=A0$(B):A1$=A1$(;1,C-1)+"#"+A1$(;C+1):A0$(B)=A1$:NEXT A:FOR
A=0 TO G:B=U(A,1):C=U(A,0):A1$=A0$(B):A1$=A1$(;1,C-1)+CHR(34)+A1$(;C+1):A0$(B)=A1$
01610 NEXT A:A1$=A0$(Y):A1$=A1$(;1,X)+"!"+A1$(;X+1):A0$(Y)=A1$
01620 INVERSE:NORMAL:RESTORE 190:A=64016:FOR A=A TO A+95:READ B:POKE A,B:NEXT A
01630 IF KEY="" THEN 1630 ELSE 360
01640 IN#0:OUT#0:CURS 23,15:PRINT"Tape Unreadable":FOR A=1 TO 1500:NEXT A:GOTO 250
```

CHASER

Chaser is for one or two players. The computer plays a fairly good game as well.

Richard Larkin
Dee Why NSW

```
00100REM CHASER

00110REM By Richard Larkin

00120 CLS : PRINT##"In this game you may play me or a friend."#"The idea of the
game is to avoid hitting anything white."+"One player uses then W,A,S,Z diamond
for movement and the"

00130 PRINT"other uses the I,J,K,M diamond."#"Any key to continue.." : I=USR(327
74)

00140 CLEAR : RESTORE : POKE162,30 : POKE163,128 : CLS : PRINT###"ONE OR TWO PLA
YERS (1 or 2)" : FOR X=1TO9999 : K1$=KEY : IF X<20 OR K1$="" THEN NEXT X ELSE IF
 K1$="2" THEN 350

00150 SD6 : CLS : PRINT## : INPUT"SKILL LEVEL 1 TO 5"H1 : IF H1<1 OR H1>5 THEN 1
50 ELSE LET H1=1-H1/10

00160 CLS : LORES : X=32 : Y=20 : A=2 : B=15 : C=1 : U=0 : D=1 : M=0 : PLOT 1,0
TO 75,0 TO 75,46 TO 1,46 TO 1,0

00170 SET X,Y : IF POINT(A,B) THEN PLAY20,3 : GOTO 250 ELSE SET A,B

00180 P=P+1 : POKE257,1 : K1$=KEY : IF K1$="A" THEN LET D=-1 : M=0 ELSE IF K1$="
S" THEN LET D=1 : M=0 ELSE IF K1$="W" THEN LET D=0 : M=1 ELSE IF K1$="Z" THEN LE
T D=0 : M=-1

00190 A=A+D : B=B+M : X=X+C : Y=Y+U : IF (X=AANDY=B) OR (X=A-DANDY=B-M) THEN PLA
Y20,3 : GOTO 250

00200 IF X<2 THEN LET X=75 ELSE IF X>75 THEN LETX=2 ELSE IF Y<1 THEN LET Y=46 EL
SE IF Y>46 THEN LET Y=1

00210 IF RND<H1 THEN 220 ELSE ON INT(RND*2+1) GOSUB 290,320

00220 IF RND<9 THEN 170 ELSE READC,U : IF U=10 THEN RESTORE : U=0

00230 GOTO 170

00240 DATA -1,0,1,0,0,1,1,0,0,-1,-1,0,0,1,1,10

00250 CURS 39,8 : PRINT"YOU HAVE BEEN HIT"

00260 CURS 39,9 : PRINT"TOTAL SCORE="P

00270 CURS 39,10 : PRINT"HIT ANY KEY TO PLAY AGAIN"

00280 I=USR(32774) : RUN

00290 IF X<A+D*3 THEN 300 ELSE LET C=-1 : U=0
12.
00300 IF X>A+D*3 THEN 310 ELSE LET C=1 : U=0

00310 IF X=A THEN 320 ELSE RETURN

00320 IF Y<B+M*3 THEN 330 ELSE LET U=-1 : C=0

00330 IF Y>B+M*3 THEN 340 ELSE LET U=1 : C=0

00340 IF Y=B THEN 290 ELSE RETURN

00350 INPUT"SPEED LEVEL (0 TO 50)"S : S=50-S : IF S<0ORS>50 THEN 350

00360 CLS : LORES : PLOT 0,0 TO 127,0 TO 127,47 TO 0,47 TO 0,0 : X=0 : Y=10 : C=
127 : U=37 : D=1 : F=-1 : H=0 : J=0 : K1$=KEY

00370 SET X,Y : SETC,U : X=X+D : C=C+F : Y=Y+H : U=U+J : IF POINT(X,Y) OR POINT(
C,U) THEN 410

00380 K1$=KEY : IF K1$="W" THEN LET H=1 : D=0 ELSE IF K1$="I" THEN LET F=0 : J=1
 ELSE IF K1$="Z" THEN LET H=-1 : D=0 ELSE IF K1$="M" THEN LET F=0 : J=-1

00390 IF K1$="A" THEN LET D=-1 : H=0 ELSE IF K1$="J" THEN LET F=-1 : J=0 ELSE IF
 K1$="S" THEN LET D=1 : H=0 ELSE IF K1$="K" THEN LET F=1 : J=0

00400 FOR T=0TOS : NEXT T : GOTO 370

00410 PLAY24,10 : IF POINT(X,Y) AND POINT(C,U) THEN LET E1$="S AT BOTH ENDS HAVE
 LOST." ELSE IF POINT(X,Y) THEN LET E1$=" ON THE RIGHT HAS WON!!!" ELSE LET E1$=
" ON THE LEFT HAS WON!!"

00420 CLS : PRINT###"THE GAME HAS ENDED AND THE PLAYER"; : SPEED255 : PRINT E1$
: SPEED0 : PRINT##"TYPE ANY KEY TO START AGAIN." : I=USR(32774) : RUN
```

JOYSTICK TEST

This program is an adaptation of testing your joystick which appeared in the Microbee Engineering Notebook. I have included the "print x print" to show your position on the screen. If you press the fire button the screen clears and you start again from your last position.

Rod Blockely
Mundingburra QLD

```
00100X=255:S=128:REM center screen
00110HIRES
00120OUT1,255:REM initialise port
00130A=IN(0):REM read joystick on port 0
00140A=143-(AAND143):REM convert to positive logic
00150B=-(AAND1):IFBTHENSETX,S:S=S+1:CURS1,1:PRINT*X*X:PRINT*S*S
00160B=-(AAND2):IFBTHENSETX,S:S=S-1:CURS1,1:PRINT*X*X:PRINT*S*S
00170B=-(AAND4):IFBTHENSETX,S:X=X-1:CURS1,1:PRINT*X*X:PRINT*S*S
00180B=-(AAND8):IFBTHENSETX,S:X=X+1:CURS1,1:PRINT*X*X:PRINT*S*S
00190B=-(AAND128):IFBTHENGOTO110
00200IFA=0THENGOTO130
00210GOTO130
```

GRAPHER

Grapher plots a two dimensional graph for values –5 to +5, and then lets you change the formula. It can also plot in three dimensions.

Richard Larkin
Dee Why NSW

```
00100 FN1=SIN(#)

00110 REM PLOTER

00120 REM By Richard Larkin

00130 POKE162,30 : POKE163,128 : CLS : PRINT↓↓↓ "            GRAPHER"♯"2 or 3 dim
ensions (Type 2 or 3)" : K1$=KEY

00140 K1$=KEY : IF K1$="2" THEN 200 ELSE IF K1$="3" THEN 150 ELSE 140

00150 CLS : PRINT♯♯ "Input step X "; INPUTS1 : INPUT "Input step Y "B1 : INPUT
"size (50 to 150)"A1

00160 ONERROR GOTO 240 : SD4 : CLS : HIRES : D1=.0327

00170 FOR H1=-A1TOA1 STEP B1 : A2=FLT( INT( .5+SQR( A1*A1-H1*H1))) : FOR B2=-A2T
OA2 STEP S1 : C2=SQR( B2*B2+H1*H1)*D1 : D2=FN1( C2) : D3=D2*20

00180 X1=B2+( H1/B1) : Y1=D3-( H1/B1) : X=INT( .75*X1) : Y=INT( .8*Y1) : SET X+2
55,Y+70 : PLOTR X+255,Y+69TOX+255,Y+60 : NEXT B2 : NEXT H1

00190 CURS 5,5 : PRINT"FINISHED" : GOTO 190

00200 CLS : HIRES : SD4 : FOR X=-100TO100 STEP 20 : FOR Y1=-100TO100 STEP 20 : S
ET 255+X,INT( Y1*.63)+128 : NEXT Y1 : NEXT X : CURS 65 : LIST 100

00210 FOR X1=-5TO5 STEP .1 : IF X1<>0 THEN LET Y1=FN1( X1) : CURS 28,14 : PRINT
X1"      ",Y1"      " : IF Y1<5.1 AND Y1>-5.1 THEN INVERT INT( X1*20+255),INT( Y1*
12.6+128)

00220 NEXT X1

00230 IF KEY="" THEN 230 ELSE POKE 220,0 : CLS : PRINT♯♯♯ "Retype The formula pr
ess 'RETURN'."♯"Then press 'RESET'" : EDIT 100

00240 CURS 5,5 : PRINT "FLUNK" : GOTO 240
```

MICROBEE

EINSTEIN II

This program was written on the Microbee and uses less than 5K! Einstein II is a game of remembering sequences as they get longer and longer. The game has five levels of difficulty. There is a three second time delay for each keystroke so you must think quickly.

The program has some unnecessary documentation which you may want to omit. It appears after line 4000. Of course all 'REMS' should not be typed. If the four keys used on lines 580 to 610 (Y,U,G,H), prove to be awkward simply alter them!

G. Adcock
Oak Park NSW

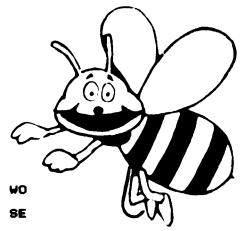
```
                ***  EINSTEIN II  ***

00010 REM****** WRITTEN BY GREG ADCOCK ******
00020 REM******     FOR PUBLIC USE     ******
00030 REM******        3/6/83          ******
00040 REM******   USES ABOUT 5K !!     ******
00050 REM*************************************
00060 REM*************************************
00100 T=0:DIMD1(40):DIMA1(40)
00110 GOTO2000
00120 T=0
00130 T=T+1
00140 PLAY 0,4
00150 J=2:K=1
00160 IF T=R THEN 750
00170 IFT>6 THENLET J=1
00180 IFT>11 THENLET J=1
00190 IFT>11 THENLET K=0
00200 G=INT(RND#5)
00210 IF G=1 THEN LET A1(T)=270
00220 IF G=2 THEN LET A1(T)=320
00230 IFG=3 THEN LET A1(T)=370
00240 IF G=4 THEN LET A1(T)= 420
00250 IF G=00RG=5 THEN 200
00260 GOTO470
00270 CLS:CURS 10,8:PCG:PRINT"AAAAAAAAAAAAAAAAAA":CURS10,5:
PRINT"BBBBBBBBBBBBBBBBB":CURS 10,6: PRINT"CCCCCCCCCCCCCCCCC
C":CURS 10,7:PRINT"DDDDDDDDDDDDDDDDD"
00280 VAR(X,Y):PLAY 2,X
00290 IFY=1 THEN PLAY 0,1
00300 CLS
00310 RETURN
00320 CLS:PCG:CURS30,5:PRINT"DDDDDDDDDDDDDDDDDD":CURS30,7:P
RINT"CCCCCCCCCCCCCCCCCC":CURS 30,8:PRINT"AAAAAAAAAAAAAAAAAA"
:CURS30,6:PRINT"BBBBBBBBBBBBBBBBB"
00330 VAR(X,Y):PLAY10,X
00340 IFY=1THEN PLAY 0,1
00350 CLS
00360 RETURN
00370 CLS:PCG:CURS10,10:PRINT"BBBBBBBBBBBBBBBBB":CURS10,11:P
RINT"CCCCCCCCCCCCCCCCC":CURS10,13:PRINT"AAAAAAAAAAAAAAAAAA"
:CURS10,12:PRINT"DDDDDDDDDDDDDDDDD"
00380 VAR(X,Y):PLAY 17,X
00390 IFY=1 THEN PLAY0,1
00400 CLS
00410 RETURN
00420 CLS:PCG:CURS30,10:PRINT"AAAAAAAAAAAAAAAAAA":CURS30,13:
PRINT"BBBBBBBBBBBBBBBBB":CURS30,12:PRINT"CCCCCCCCCCCCCCCCC
":CURS30,11:PRINT"DDDDDDDDDDDDDDDDD"
00430 VAR(X,Y):PLAY 24,X
00440 IF Y=1 THEN PLAY 0,1
00450 CLS
00460 RETURN
00470 W=0
00480 W=W+1
00490 IF W>T THEN 520
00500 GOSUBIJ,K]INT(A1(W))
00510 GOTO480
00520 Q=0
00530  Q=1+Q
```

MINDER

Minder is like master mind except you can choose the number of columns, colours and guesses you get.

Richard Larkin
Dee Why NSW

```
00100REM MINDER

00110REM By Richard Larkin

00120 CLS : PRINT#"Welcome to Minder"#"You will be asked how many columns you wi
sh then colours"#"(represented by letters A-Z) then how many guesses."

00130 PRINT"You must then type in your guess for the computers hiden code."#"Bac
k space may be used."#"The rules are the same as for Master-mind."##"Any key to
start..." : I=USR(32774)

00140 POKE220,63 : INVERSE : CLS : POKE162,30 : POKE163,128 : CURS 26,16 : PRINT
"MASTER-MIND."; : NORMAL : FOR X=1TO7 : FOR Y=1TO60+X*X*3 : NEXT Y : PRINT : NEX
T X

00150 POKE220,0 : CLEAR : INPUT"How many columns would you like (1 to 13) ?"I7$
: C=INT(VAL(I7$)) : IF C>13 OR C<1 THEN 150

00160 INPUT"How many different colours (2 to 26) ?"I7$ : L=INT(VAL(I7$)) : IF L>
26 OR L<2 THEN 160

00170 INPUT"How many guesses would you like (1 to 13) ?"I7$ : G=INT(VAL(I7$)) :
IF G<1 OR G>13 THEN 170

00180 DIM C1(C),G1(C),W1(C) : D=0 : CLS : UNDERLINE : CURS26,1 : PRINT"Master-Mi
nd." : NORMAL : FOR X=1TOC : C1$(X)=CHR(INT(RND*FLT(L)+65)) : NEXT X
```



```
00540 IF@=T+1 THEN GOTO 130
00550 S=0
00560 S=S+1:IF S=420 THEN 740
00570 K1@=KEY:IFK1@="" THEN560
00580 IFK1@="Y" THEN680
00590 IF K1@="U" THEN 660
00600 IF K1@="D" THEN 700
00610 IFK1@="H" THEN 720
00620 GOTO560
00630 REM
00640 REM
00650REM
00660 R1=320:IFR1=A1(@)THENGOSUB[J,K]320
00670 R1=320:IF R1(A1(@) OR R1)A1(@) THEN 740ELSEGOTO530
00680 R2=270:IFR2=A1(@) THEN GOSUB[J,K]270
00690 R2=270:IF R2(A1(@)ORR2)A1(@) THEN 740ELSEGOTO530
00700 R3=370:IF R3=A1(@) THEN GOSUB[J,K]370
00710 R3=370:IF R3)A1(@)OR R3(A1(@)THEN740ELSEGOTO 530
00720 R1=420:IFR1=A1(@) THENGOSUB[J,K]420
00730 R1=420:IF R1(A1(@)OR R1)A1(@) THEN 740ELSEGOTO 530
00740 NORMAL:PRINT"WRONG":PLAYO,8:GOTO760
00750 NORMAL:T=R-1: GOTO 3000
00760 IF T>=E THEN 780
00770 NORMAL:GOTO820
00780 E=T
00790 FOR I=1 TO W
00800 D1(I)=A1(I)
00810 NEXT I
00820 NORMAL:INPUT"  #### '1' TO CONTINUE...'2' TO SEE BEST SCORE... OR '3' TO CHANGE  LEVELS...  ####" P1@:CLS:IFP1@="1" THEN GOTO930
00830 IFP1@="2" THEN870
00840 IFP1@="3"THEN860
00850 GOTO820
00860 GOTO2030
00870 T=0:I=0
00880 I=I+1
00890 PRINT I
00900 GOSUB[I,1]INT(D1(I))
00910 IFI)E-1THEN 820
00920 GOTO880
00930 T=0: GOTO130
01000 P=63488+65*16
01010 FOR A=P TO P+16*4-1
01020 READB:POKEA,B
01030 NEXTA
01040 DATA 248,248,248,248,248,248,248,248,248,248,248,248,248,248,248,248
01050 DATA 7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7
01060 DATA 6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6
01070 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
01080 GOTO2030
01090 CLS:NORMAL
02000 NORMAL:CLS
02010 PRINT[A63 144]\\"   MIKE ROBEE PRESENTS ...."\"
          EINSTEIN(II)"\\[A63 144]
02020 GOTO 4000
02030 INPUT"WHICH LEVEL WOULD YOU LIKE,1(EASIEST),2,3,4,5 OR A RANDOM NUMBER(X)."W1@
02040 CLS
02050 IFW1@="1" THEN LET R=7
02060 IFW1@="2"THEN LET R=10
02070 IFW1@="3"THENLETR=14
02080IFW1@="4"THENLETR=21
02090IFW1@="5"THENLETR=33
02100IFW1@="X"ORW1@="x"THEN LETR=INT(RND#29)
02110 GOTO120
03000 CURS 20,8:PLAY 16,16,20,23,0,20,23,8:PRINT" YOU'VE  WON  !!"
03010 NORMAL:CURS8,9:PRINT" AND I'M JOLLY WELL PLEASED TO SEE IT TOO."
03020 PC@
03030 GOTO760
04000 PRINT"Hello,"
04010 PRINTSPC(7)"I would like to tell you a few things about this program"\"to make it easier to run. It is not an easy  game so don't be"\"deceived."
04020 FORI=1 TO 3000:NEXTI:PLAY10,4
04030 PRINTSPC(7)"It is much like the more expensive game 'EINSTEIN' (only"\"much better!!!) "
04040FORI=1 TO 2600:NEXTI:PLAY 10,4
04050 PRINTSPC(7)"If you already know about this game, then press 'S'"\"to get started. Press 'I' to receive more information."
04060 FORI=1TO2500:NEXTI:PLAY15,4
04070 LI@=KEY:IFLI@="" THEN 4070
04080 IF LI@="S" THEN 1000
04090 IF LI@="I"THEN 4110
04100 GOTO4070
04110 NORMAL:CLS:INVERSE:PRINT"THERE ARE 5 LEVELS OF PLAY ;1 BEING THE EASIEST"\"AND 5 THE HARDEST."
04120LETA3@="REPETITIONS":NORMAL:PLAYO,20:CURS 14,3:PRINT"1 =     SIX   "+A3@
04130CURS14,5:PRINT"2=      NINE   "+A3@
04140CURS14,7:PRINT"3=      THIRTEEN  "+A3@
04150CURS14,9:PRINT"4=      TWENTY   "+A3@
04160 CURS14,11:PRINT"5=      THIRTY-TWO   "+A3@
04170 CURS14,13:PRINT"X=      RANDOM NUMBER CHOSEN BY THE COMPUTOR,":CURS 14,14:PRINT"RANGING FROM 1 TO 28!!!"
04180 CURS30,16:INPUT"PRESS(RETURN)"G4@
04190 IFG4@="" THEN4200
04200 CLS:PRINT"  The object of the game is to successfully repeat the tune"\"and pattern the computor gives you.There are four different"4"\" notes each with its own 'square'"
04210 PRINT" and you must try to "\"match them using the following keys..."
04220 PRINT:PRINT"                    W   E   R   T "|:INVERSE:PRINT" Y"|:NORMAL:PRINT"  "|:INVERSE:PRINT"U "|:NORMAL:PRINT" I   O   P"
04230PRINT:PRINT"                      S   D   F "|:INVERSE:PRINT" G "|:NORMAL:PRINT"  "|:INVERSE:PRINT"H "|:NORMAL:PRINT" J   K   L  "
04240PLAYO,45,12,4:PRINT:PRINT" THIS IS NOT ALL THOUGH."
04250 CURS30,16:INPUT"PRESS (RETURN)"Y4@
04260IF Y4@="" THEN 4270
04270CLS:PLAY14,4:PRINT" You also have to reply to each note within a time limit!"\"This limit is between 3 & 4 seconds- so don't relax!!!"
04280 PRINT "The game also gets faster the further you get into it."\" SO THIS IS MIKE ROBEE WISHING YOU GOOD LUCK IN YOUR VENTURE...."
04290 INPUT"                    (RETURN) TO START..."K6@
04300 IF K6@="" THEN 1000
```

```
00190 S=25-(C*3)/2 : A=S+C*3 : CURS A,2 : PRINT"Black  White" : FOR X=STOS+C*3-3 STEP3 : CURS X,2 : PRINT"?" : CURSX,3 : PRINT"#" : FOR Y=4TO3+G : CURS X,Y : PRINT"."; : NEXT Y : NEXTX

00200 U=1 : B=0 : W=0 : K1$=KEY : CURSS+U*3-4,D+4 : PRINT" ";

00210 POKE257,1 : K1$=KEY : IF ASC(K1$)=8 AND U)1 THEN LET U=U-1 : CURSS+U*3-3,D+4 : PRINT"."; : CURSS+U*3-4,D+4 : PRINT" "; : GOTO 210 ELSE IF K1$("A" OR K1$)CHR(L+64) THEN 210

00220 G1$(U)=K1$ : CURS S+U*3-3,D+4 : PRINT K1$; : U=U+1 : CURS S+U*3-4,D+4 : PRINT" "; : IF U(C+1 THEN 210 ELSE FOR X=1TOC : IF C1$(X)=G1$(X) THEN LET B=B+1

00230 NEXT X : IF B=C THEN 250 ELSE FOR X=1TOC : W1$(X)=C1$(X) : NEXT X : FOR X=1TOC : FOR Y=1TOC : IF G1$(X)()W1$(Y) THEN NEXT  Y ELSE LET W=W+1 : W1$(Y)="" : NEXT*Y 240

00240 NEXT X : W=W-B : CURS A+1,D+4 : PRINT B; : CURS A+8,D+4 : PRINTW; : D=D+1 : IF D(G THEN 200

00250 POKE220,63 : FOR X=1TOC : CURS S+X*3-3,2 : PRINT C1$(X) : PLAYINT((FLT(ASC(C1$(X)))-64)/1.13)+1 : NEXT X : I=USR(32774) : CLS : PRINT+++

00260 IF B=C THEN PRINT"Congratulations you have guessed the secret code !" ELSE PRINT"Sorry you have not guessed correct."

00270 PRINT"Type any key to play again." : I=USR(32774) : CLS : PRINT### : GOTO 150
```

KEY CLICK

The program simply calls up a machine code routine every time a character is printed. This is a very useful program for typing or whenever a character is printed such as a BASIC error or listing a program. It will produce an audible signal for every character printed.

All the program does is print the relevant character for the key pressed then produce a sound. To change the tone and length of the note alter the 90 in line 140 for length and the 20 for the tone.

I find it much better for typing to hear a nice sharp 'beep' rather than a 'clunk'.

Alistair Ferrier
Coleraine Vic

```
00100 FOR A=15000 TO 15022
00110 READ B
00120 POKE A,B
00130 NEXT A
00140 DATA 71,205,12,128,6,90,14,20,62,0,211
00150 DATA 2,13,32,253,62,255,211,2,5,32,240,201
00160 REM     Change output vector to jump to MC routine
00170 REM     every time a character has to be printed.
00180 POKE 178,152:POKE 179,58
00190 REM
00200 REM    To restore type "POKE 178,47:POKE 179,166"
```


PASSWORD

This little routine discourages unauthorized activity.

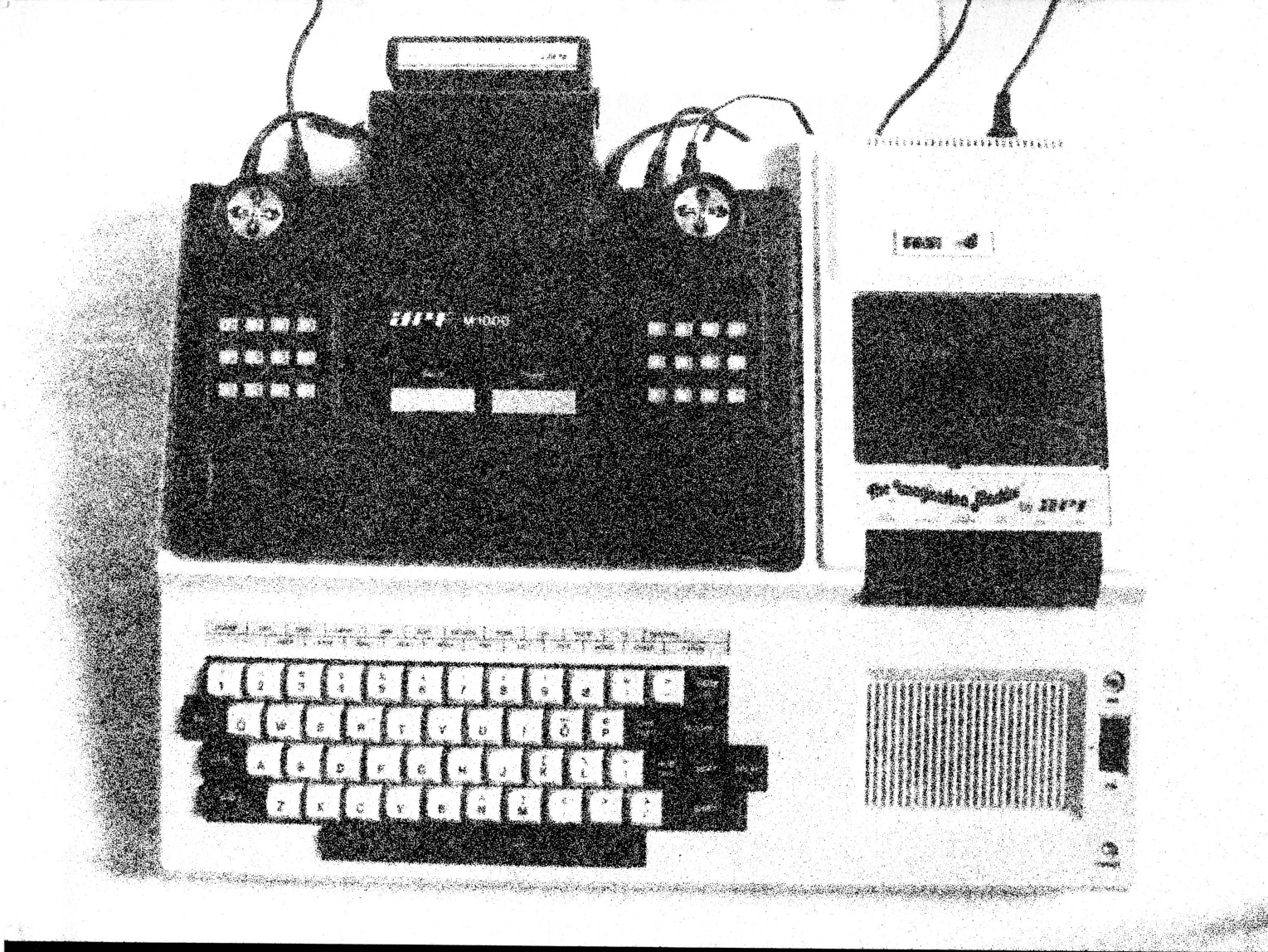
On violation of this 'Password' sub-routine the singular features of the APF cause the keyboard to be disabled.

BREAK will terminate the audio alarm (Line 9030) but entry to the high resolution mode in this fashion (Line 9020) permanently locks out keyboard response. The only way out is to power down and re-load the program.

Obviously you would not use this approach unless you can retrieve from disc or tape what you were working on. The catastrophic effects of a password violation will permanently discourage fiddly fingers however!

J. Elkhorne
Chigwell Tas

```
1  GOTO 100
9  REM   GOTO MAINLINE
10  REM      -- PASSWORD -- J.L. ELKHORNE
20  DIM I$(1): DIM A$(1): DIM C$(99)
25  REM   C$ IS PASSWORD
30 I$= KEY$ (0): IF I$="" THEN 30
40 A$=I$:L= LEN (C$)
50 I$= KEY$ (0):  IF I$<>"" THEN 50
60 C$(L)=A$
70  IF C$(0)="MOOSE" THEN  RETURN
80  IF L>4 THEN 9000: REM   L TIED TO PASSWORD USED
90  GOTO 30
100  REM   MAINLINE
110  REM   TEST SUBROUTINE
120  PRINT "PASSWORD?": GOSUB 10
130  PRINT "SUCCESS!": STOP
9000  PRINT "SECURITY VIOLATION!"
9010  FOR X=1 TO 100: NEXT X
9020  POKE 8193,60: POKE 8194,158
9030  CALL 17920: GOTO 9030
```



PROGRAMS FOR APF IMAGINATION MACHINE

MISSION IMPOSSIBLE

You are flying a bomber over an enemy installation and your fuel is running low. You must try to land – and to do so must bomb most of the buildings below you that are in your way. If you hit a building your plane explodes, if you run out of fuel you crash. Beware of enemy gunfire as well.

The program uses the keys: A, Z and 1,2,3 (on the keypad only). If you want to generate your own landscape you must give the heights (1-9) of each of the 32 buildings. They are drawn as you go.

The machine program (given by the data) generates 16 bytes (which are stored in locations 32 to 47). If you record what happens when a particular key is pressed (or a combination of keys!) you will find which bytes must be checked for different keys. (Note that the data given above is the first 17 bytes of the machine program given).

If you think that the game as given is too easy or hard then change line 420 to suit (that is, increase/decrease the number of clusters or forward fire left or the fuel allowed).

Peter Fallon

Peter, send us your address!

```
10 REM Mission Impossible,
15 REM written by Peter Fallon.
49 REM Data for plane tail,CHR$(192)
50 DATA 192,240,252,255,255,255,255,255
54 REM Plane body,CHR$(193)
55 DATA 0,0,0,0,255,255,255,255
59 REM Plane nose,CHR$(194)
60 DATA 0,0,0,0,192,240,252,255
64 REM Building,CHR$(195)
65 DATA 255,255,255,255,255,255,255,255
69 REM Base,CHR$(196)
70 DATA 170,170,85,85,170,170,85,85
74 REM Bomb left,CHR$(197)
75 DATA 2,2,1,3,7,7,3,1
79 REM Bomb right,CHR$(198)
80 DATA 64,64,128,192,224,224,192,128
84 REM Explosion,CHR$(199)
85 DATA 204,204,51,51,204,204,51,51
89 REM Bullet left,CHR$(200)
90 DATA 0,0,1,3,3,1,0,0
94 REM Bullet right,CHR$(201)
95 DATA 0,0,128,192,192,128,0,0
99 REM Gun left,CHR$(202)
100 DATA 1,3,7,15,31,255,255,255
104 REM Gun right,CHR$(203)
105 DATA 128,192,224,240,248,255,255,255
109 REM B(5,2) holds the location of the enemy guns
110 RESTORE : DIM B(5,2)
120 FOR Z = -512 TO -417 : READ A : POKE Z,A : NEXT
130 PRINT CHR$(12);TAB(20);"┌---------------┐"
135 PRINT          TAB(20);"|  Mission      |"
140 PRINT          TAB(20);"|  Impossible   |"
145 PRINT          TAB(20);"└---------------┘"
160 PRINT : PRINT TAB(40);"By Peter Fallon"
170 PRINT : PRINT"Land your plane Before you run out of fuel."
180 PRINT : PRINT"Bomb out the buildings in your way"
190 PRINT : PRINT"and avoid enemy fire!"
200 PRINT : PRINT"Use the keys :"
210 PRINT : PRINT"  A          for up"
220 PRINT : PRINT"  Z          for down"
225 PRINT : PRINT" And the keypad numbers :"
230 PRINT : PRINT"  1          to drop a bomb"
240 PRINT : PRINT"  2          to drop a cluster bomb"
250 PRINT : PRINT"  3          to fire forward guns"
260 PRINT : PRINT"Do you want bomb trajectory to be (A)ngled or"
261 PRINT : PRINT"(S)traight ?"
265 REM Data for  GET $ routine
269 DATA 205,21,224,194,250,223
270 DATA 205,9,224,50,255,0,201
275 FOR Z = 240 TO 252 : READ A : POKE Z,A : NEXT
280 POKE 260,240 : POKE 261,0 : Z=USR(0)
281 IF PEEK(255) = 0 THEN 280
282 IF PEEK(255) = 83 THEN T=0 : GOTO 290
283 IF PEEK(255) = 65 THEN T=2 : GOTO 290
285 GOTO 280
290 GOSUB 1400
300 IF AA=1 THEN 370
310 PRINT CHR$(12);
315 REM Set random buildings
320 FOR Z = 0 TO 31
330 A=INT(RND(1)*10+1)
340 FOR W = 28 TO 28-A STEP-1
350 J=Z*2+W*64-3968 : POKE J,195 : POKE J+1,195
360 NEXT : NEXT
365 REM Set base
370 FOR Z = 0 TO 63 : POKE Z-2112,196 : NEXT
375 REM Set random location of enemy guns
380 FOR Z = 1 TO 5
```

```
390 A=INT(RND(1)*32) : B(Z,1)=A*2
400 IF PEEK(A*2-2112)=202 THEN 390
405 POKE A*2-2112,202 : POKE A*2-2111,203
410 NEXT
415 REM Set variables : X,Y=plane co-ords;FU=fuel left
416 REM CC=clusters left;FG=forward fire left
420 X=2 : Y=1 : A=0 : B=0 : CC=5 : FG=5 : D=0 :
    FU=300 : EF=0 : G=0
430 J=X+Y*64-3968 : POKE J-2,192 : POKE J-1,193
440 POKE J,193 : POKE J+1,194
445 REM Start game : Scan keyboard
450 POKE 260,0 : POKE 261,0 : Z=USR(0)
460 IF PEEK(255)=0 THEN 700
470 ON PEEK(255) GOTO 490,500,510,520,530
490 A=-1 : FU=FU-10 : GOTO 700
500 A=1 : GOTO 700
510 IF B=1 THEN 700
515 B=1 : GOTO 620
520 IF B=1 THEN 700
521 IF CC=0 THEN 700
522 B=1 : CC=CC-1 : C=1
525 GOTO 620
530 IF FG=0 THEN 700
535 FG=FG-1 : XX=X+2 : YY=Y
536 IF XX=64 THEN XX=0 : YY=YY+1
540 IF YY=29 THEN YY=28
550 FOR Z = 1 TO 10
555 J=XX+YY*64-3968
560 POKE J,45 : POKE J+1,45
580 POKE J,32 : POKE J+1,32
590 XX=XX+2 : IF XX=64 THEN XX=0 : YY=YY+1
600 IF YY=29 THEN YY=28
610 NEXT : GOTO 700
620 IF Y<>28 THEN 670
630 FOR Z = 1 TO 5
640 IF B(Z,1)=X AND B(Z,2)=0 THEN B(Z,2)=1 :
    EG=EG+1 : GOTO 660
650 NEXT : B=0 : C=0 : GOTO 700
660 POKE X-2112,32 : POKE X-2111,32 : B=0 : C=0 :
    GOTO 700
670 J=X+(Y+1)*64-3968 : IF J=195 THEN D=1
680 E=X : F=Y+1
690 POKE J,197 : POKE J+1,198
700 J=X+Y*64-3968 : POKE J,32 : POKE J+1,32
710 POKE J-2,32 : POKE J-1,32 : POKE 255,0
720 X=X+2 : IF X=64 THEN X=0 : Y=Y+1 : FU=FU-10
725 Y=Y+A : A=0
730 IF Y 28 THEN 750
740 Y=28 : G=G+1 : IF G=32 THEN 1380
750 J=X+Y*64-3968 : IF PEEK(J)=195 THEN POKE J,199 :
    POKE J+1,199 : GOTO 1260
760 POKE J-2,192 : POKE J-1,193
770 POKE J,193 : POKE J+1,194
780 PRINT CHR$(17);"Fuel left =";FU;"
    Clusters left =";CC;
785 PRINT" Forward fire left =";FG
790 IF FU=0 THEN 1200
800 IF EF=1 THEN 850
810 EF=1 : IF EG=5 THEN EF=0 : GOTO 910
820 W=INT(RND(1)*5+1) : IF B(W,2)=1 THEN 820
830 W=B(W,1) : U=28
840 GOTO 870
850 J=W+U*64-3968 : POKE J,32 : POKE J+1,32
860 U=U-1
870 IF U>Y THEN EF=0 : GOTO 910
890 IF W=X AND U=Y THEN 1260
900 J=W+U*64-3968 : POKE J,200 : POKE J+1,201
910 IF B=0 THEN 450
920 J=E+F*64-3968 : POKE J,32 : POKE J+1,32
930 F=F+1 : IF F 29 THEN 990
940 FOR Z = 1 TO 5
950 IF B(Z,1)< > E OR B(Z,2)=1 THEN 970
960 POKE E-2112,32 : POKE E-2111,32 : B(Z,2)=1 :
    EG=EG+1
970 NEXT : B=0 : IF C=0 THEN 450
980 F=F-3 : GOTO 1020
990 J=E+F*64-3968 : IF PEEK(J)=195 THEN 1020
1000 POKE J,197 : POKE J+1,198
1010 GOTO 450
1020 J=E+F*64-3968 : POKE J,199 : POKE J+1,199
1030 POKE J,32 : POKE J+1,32
1040 IF C=1 THEN GOSUB 1060
1050 F=F+1 : D=D+1 : IF F=28 OR D=4 THEN B=0 : C=0 :
     D=0 : GOTO 450
1060 IF E=0 THEN 1100
1070 POKE J-2,199 : POKE J-1,199
1080 POKE J-2,32 : POKE J-1,32
1090 IF E=62 THEN RETURN
1100 POKE J+3,199 : POKE J+2,199
1110 POKE J+3,32 : POKE J+2,32
1120 RETURN
1195 REM Crash and explode routine
1200 J=X+Y*64-3968 : POKE J-2,32 : POKE J-1,32
1210 POKE J,32 : POKE J+1,32
1220 Y=Y+1 : IF PEEK(J+64)>194 then 1260
1230 POKE J+62,192 : POKE J+63,193
1240 POKE J+64,194 : POKE J+65,194
1250 GOTO 1200
1260 FOR Z = Y-2 TO Y+2
1270 FOR W = X-2 TO x+2
1280 POKE W+Z*64-3968,210+INT(RND(1)*5+1)
1290 NEXT : NEXT
1300 FOR Z = 1 TO 100
1310 POKE -368+INT(RND(1)*40+1),INT(RND(1)*256)
1320 NEXT : PRINT : PRINT"You lost!!"
1330 PRINT : PRINT"Do you want to play again ? Y/N"
1340 POKE 260,240 : POKE 261,0 : Z=USR(0)
1350 IF PEEK(255)=0 THEN 1340
1360 IF PEEK(255)=89 THEN RUN
1370 END
1380 PRINT : PRINT"You landed!!  Congratulations!!"
1390 GOTO 1330
1395 REM Routine to let you make your own landscape
1400 PRINT : PRINT"Do you want (R)andom buildings or
     (Y)our own?"
1410 POKE 260,240 : POKE 261,0 : Z=USR(0)
1420 IF PEEK(255)=82 THEN AA=0 : RETURN
1430 IF PEEK(255)=89 THEN 1440
1435 GOTO 1410
1440 PRINT CHR$(12);"Give height  (1-9)"
1450 FOR Z = 0 TO 31
1460 POKE 260,240 : POKE 261,0 : Z=USR(0)
1470 IF PEEK(255)=0 THEN 1460
1480 IF PEEK(255)<49 OR PEEK(255)>57 THEN 1460
1490 A=PEEK(255)-49
1530 FOR W = 28 TO 28-A STEP-1
1540 J=Z*2+W*64-3968 : POKE J,195 : POKE J+1,195
1550 NEXT : NEXT
1560 AA=1 : RETURN
```

```
0000:01 FE 10   LD BC,10FE      ;Keyboard scan routine
0003:21 2F 01   LD HL,012F      ;Results stored in loca
0006:ED 41      OUT (C),B       ; 0100-012F.       tions
0008:ED 78      IN A,(C)
000A:F6 E0      OR E0
000C:2F         CPL
000D:77         LD (HL),A
000E:2B         DEC HL
```

Sorcerer

MISSION IMPOSSIBLE

```
000F:10 F5      DJNZ F5
0011:3A 21 01   LD A,(0121)     ;is it 'A' (up)?
0014:FE 04      CP 04
0016:C2 1F 00   JP NZ,001F      ;no,try again
0019:3E 01      LD A,01         ;yes,save itin location FF (255)
001B:32 FF 00   LD (00FF),A
001E:C9         RET             ;return to basic program
001F:FE 02      CP 02           ;is it 'Z' (down)?
0021:C2 2A 00   JP NZ,002A      ;no try again
0024:3E 02      LD A,02         ;yes save it
0026:32 FF 00   LD (00FF),A
0029:C9         RET             ;return
002A:3A 2C 01   LD A,(012C)     ;is it '1' on keypad (drop bomb)?
002D:FE 02      CP 02
002F:C2 38 00   JP NZ,0038      ;no try again
0032:3E 03      LD A,03         ;yes save it
0034:32 FF 00   LD (00FF),A
0037:C9         RET             ;return
0038:3A 2D 01   LD A,(012D)     ;is it '2' on keypad (drop cluster)?
003B:FE 02      CP 02
003D:C2 46 00   JP NZ,0046      ;no try again
0040:3E 04      LD A,04         ;yes save it
0042:32 FF 00   LD (00FF),A
0045:C9         RET             ;return
0046:3A 2E 01   LD A,(012E)     ;is it '3' on keypad (fire guns)?
0049:FE 10      CP 10
004B:C0         RET NZ          ;no,return
004C:3E 05      LD A,05         ;yes save it
004E:32 FF 00   LD (00FF),A
0051:C9         RET             ;end of program,return
```

SPIRO FOR MBASIC

Spiro is a program that draws patterns similar to those produced by the well-known 'Spirograph' game. It will draw all patterns that use two wheels, with the second wheel either inside or outside the first one. In addition, it allows the pen radius in the second wheel to be outside the circumference of the wheel.

It is written in Microsoft BASIC-80 (MBasic) but does not use any special commands, so should be easily portable. Like most Basics, MBasic does its transcendental functions in radians, so the program has a conversion function (ANGLE).

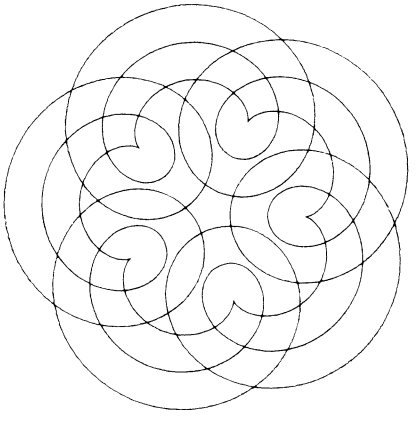
It uses only two commands in the plotter. 'MX,Y' means move to position X,Y and 'DX,Y means draw a line from the current pen position to position X,Y. The pattern will be centered 600 units along the X and Y axes, but this can be changed with one line in the program.

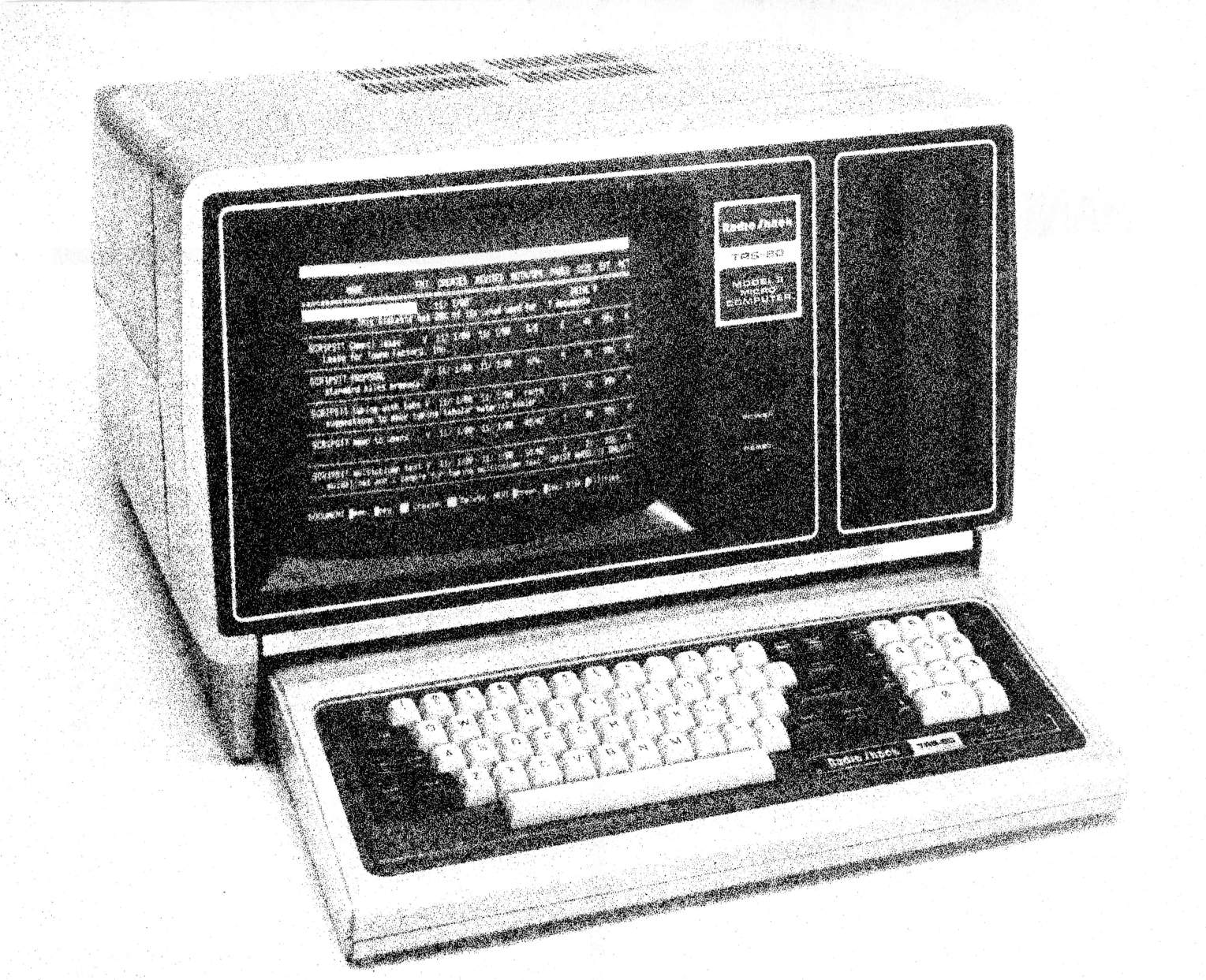
```
10 REM SPIRO
20 CLS$=CHR$(26):PRINT CLS$;  CLEAR SCREEN
30 PRINT"This program draws circular patterns similar to those produced
40 PRINT"by the ";CHR$(34);"SPIROGRAPH";CHR$(34);" game. The plotting
       commands are
50 PRINT"suitable for the ROLAND DG DXY-100 and DXY-800 series of plotters.
70 PRINT"The program requires 5 items of input.
80 PRINT" 1. Large Circle Radius. This is the radius of the gear wheel that
90 PRINT"     normally would be pinned to the drawing surface.
110 PRINT" 2. Small circle radius. This is the radius of the gear wheel
       into which
120 PRINT"     the pen is inserted and which rolls around the circumference of
130 PRINT"     the large circle. Despite the names, the small circle radius can
140 PRINT"     be either smaller or larger than the large circle. For practical
150 PRINT"     purposes the radius can be considered as the number of teeth
       in the
160 PRINT"     circumference. Notice that the ratio of large circle radius to
170 PRINT"     small circle radius determines the number of iterations needed
180 PRINT"     before a pattern returns to its starting point.
190 PRINT"     Note that the small circle radius can be negative. This means
200 PRINT"     that the small circle rolls around an inside circumference. In
210 PRINT"     this case the pen position radius must be less than the large
220 PRINT"     circle radius.
230 PRINT"HIT ANY KEY TO CONTINUE";I$=INPUT$(1)
240 PRINT CLS$;
250 PRINT" 3. Pen position radius. This is the distance from the centre of the
260 PRINT"     small circle at which the pen will be placed. This can be
       greater
270 PRINT"     than the radius of the small circle, to create effects not
       possible
280 PRINT"     with the original game.
290 PRINT" 4. Number of iterations. The number of times that the small
       circle is
300 PRINT"     to complete a circuit of the large circle.
310 PRINT" 5. Initial offset. The angle by which the radius vector containing
320 PRINT"     the pen is to be rotated from the x-axis (0 degrees) at
       the start
330 PRINT"     of the plot. This is used in repeated patterns to 'walk'
       the whole
340 PRINT"     pattern around. For single patterns, use 0.
350 PRINT:PRINT
390 PRINT"HIT ANY KEY TO START";I$=INPUT$(1)
400 PRINT CLS$;
1000 ANGLE=3.1415972000000001#/180
1010 INPUT " LARGE CIRCLE RADIUS = ",R0
1020 INPUT " SMALL CIRCLE RADIUS = ",R1
1030 INPUT " PEN POSITION RADIUS = ",R
1040 INPUT "NUMBER OF ITERATIONS = ",N
1050 INPUT "      INITIAL OFFSET = ",T
1060 DISPL=600 'OFFSET. ADJUST FOR YOUR PLOTTER OR PATTERN SIZE
1080 F=0
1090 FOR TH=0 TO 360*N
1100  CX=(R0+R1)*COS(TH*ANGLE):CY=(R0+R1)*SIN(TH*ANGLE)
1110  SI=TH:IF R1<>0 THEN SI=TH*(R0+R1)/R1
1120  IF SI>360 THEN SI=SI-360:GOTO 1130
1130  PX=CX+(R*COS((T+SI)*ANGLE)):PY=CY+(R*SIN((T+SI)*ANGLE))
1140  PX=PX+DISPL:PY=PY+DISPL
1150  X1=INT(PX):Y1=INT(PY):GOSUB 1200
1160 NEXT TH
1170 GOTO 400
1200 IF F=0 THEN LPRINT "M";X1;",";Y1 'MOVE TO (X1,Y1)
1210 F=1
1220 LPRINT "D";X1;",";Y1 'DRAW TO (X1,Y1)
1230 RETURN
```

Enhancements include changing the shape of the 'big' circle to an ellipse or some other function or inserting an auto-incrementing loop to rotate a pattern. It would also be possible to add another 'circle' to the structure. If R2 is the radius of this circle, then calculate SJ as for SI, but from R1 and R2, and then calculate QX and QY as for PX and PY.

The pattern here was produced with the sequences 250/50/250/1/0,250/50/150/1/0 and 250/50/50/1/0.

Jeff Richards
Jamboree Heights Qld





PROGRAMS FOR TRS-80

HANGMAN

Hangman should work on the Model I and, with small modifications, any other machine. The normal rules of hangman apply, however, if requested, there is a time limit applied to each guess. If a key is not pressed in time a move will be lost. As you win or lose the time limit changes to make it more difficult or easier.

Obviously many more words can be introduced by simply adding more data lines, between lines 180 and 210. SE$ contains the hidden word and DI$ contains the displayed word, for example RA--I-. TI contains the time limit and its initial value can be altered by editing line 60.

R. Tooth
Devonport TAS

```
10 PRINT "HANGMAN","By R.Tooth 1982"
20 PRINT "-------"
30 PRINT "ANY KEY TO CONTINUE"
40 IF INKEY$="" THEN 40
50 PRINT"DO YOU WANT A TIME LIMIT (Y/N)"
60 Y$=INKEY$:IF Y$="Y" THEN TI=800 ELSE IF Y$="N"
THEN TI=90000 ELSE 60
70 RESTORE:ON ERROR GOTO 70
80 T=0:Y=0:H=0:FOR LP=1 TO RND(30)
90 READ SE$      ' SE$ CONTAINS HIDDEN WORD
100 NEXT LP
110 CLS          ' SET UP SCREEN
120 PRINT @ 136,CHR$(175)
130 PRINT @ 137,STRING$(12,143)
140 FOR LP=1 TO 9:PRINT @ 148+(LP-1)*64,CHR$(191);
150 NEXT LP
160 PRINT @ 718,STRING$(12,143);
170 Y=1:H=64:PRINT @ 200,CHR$(138);
180 DI$=STRING$(LEN(SE$),"-")    ' DI$ CONTAINS DIS
PLAYED WORD
190 DATA BLACK,HELLO,CHAIR,DISCOVERY,IMAGINATION,T
OE,OPERATION,LANGUAGE,EQUIPMENT,ASK,AGENCY,AQUIRED
,IF,QUARTZ,SHALLOW,DYE,FRIGHT,PITCH,CUBICLE,TURTLE
,LION,CAT,INCREDIBLE,RABBIT,YACHT,TELEPHONE
200 DATA CLOCK,STAPLER,QUILT,INGENIOUS,IT,TYRE,HAP
PY,HAND,WICK,MISSIPI,WINDOW,PSYCHIATRIST,WITHDRAWL
,OBJECTIVE,FIX,LOWER,COMPLEX,SENSE,ELECTRICITY,HIG
H,QUADRANT,TWO,AND,YOGHURT,QUICKLY,LABORATORY,YEST
ERDAY,BLIMP,HYDROGEN,CHEQUE,SKI,AQUA,BOG
210 PRINT @ 900,DI$;
220 IF DI$=SE$ THEN GOTO 450  ' HIDDEN WORD DISCOV
ERED
230 PRINT @ 964,"";
240 FOR LP=1 TO TI:C$=INKEY$:IF C$("[" AND C$)"@"
THEN 270
250 NEXT LP
260 C$=" "
270 L$=C$+","
280 PRINT @ 970+T,L$;
290 F=0:T=T+2
300 IF INSTR(1,SE$,C$)=0 OR INSTR(1,DI$,C$)()0 THE
N 350
310 F=INSTR(F+1,SE$,C$)    ' F CONTAINS POSITION OF
 INPUTTED KEY IN WORD
320 IF F=0 THEN GOTO 210
330 MID$(DI$,F,1)=C$
340 GOTO 310
350 IF Y=7 THEN H=128:FOR LP=0 TO 9:PRINT @ 260+LP
*64,"          ";:NEXT LP:FOR J=1 TO 2:PRINT @ 524+
J*64,CHR$(191);:NEXT J:PRINT @ 264,CHR$(138);:PRIN
T @ 200,CHR$(170);:PRINT @ 4,"Y O U R   H A N G E
D!":TI=TI+8*Y:PRINT@900,SE$;  '  HANGED
360 IF Y)0 THEN PRINT @ 199+H,"('')";
370 IF Y)1 THEN PRINT @ 263+H,CHR$(156);CHR$(191);
CHR$(157);CHR$(148);
380 IF Y)2 THEN PRINT @ 326+H,CHR$(136);CHR$(133);
CHR$(191);CHR$(149);CHR$(141);
390 IF Y)3 THEN PRINT @ 391+H,CHR$(168);CHR$(143);
CHR$(173);
400 IF Y)4 THEN PRINT @ 455+H,CHR$(142);" ";CHR$(1
38);CHR$(132);
410 IF Y=7 THEN 470
420 IF Y)5 THEN PRINT @ 517+H,STRING$(8,131);
430 Y=Y+1       ' Y CONTAINS NUMBER OF INCORRECT GU
ESSES
440 GOTO 220
450 PRINT:PRINT "CONGAGULATIONS!! YOU GOT IT"
460 TI=TI-(10-Y)*10      ' ADJUST TIME FACTOR
470 PRINT @ 1000,"PLAY AGAIN?";
480 B$=INKEY$:IF B$="" THEN 480  ELSE IF B$()"N" T
HEN 80
490 PRINT:PRINT "HEARTLESS BEAST"
500 END
```

SHUFFLE

Here's a fun little program to keep you and your mind occupied on rainy Saturday afternoons or on that long business trip interstate. Written for the Model 100 TRS-80, it is a number-shuffle game which scrambles a line of digits and dares you to put them in their correct order. Ten digits are placed, in jumbled order, in a line. Your job is to 'shuffle' them by reversing part of the sequence, eventually to have all ten digits in the order 0123456789.

After enduring the title pages and introductory instructions (housed in the subroutine from line 260 onwards) you are greeted with a rather sparse screen holding only three pieces of data – a counter stating what move you are up to, the jumbled line of numbers and an input prompt asking you which digit you'd like to reverse from. When you enter a number between 1 and 10, the sequence from that number to the end is totally reversed. For example, if you enter 4 then the first four numbers will remain as they are and the rest of the line will reverse its sequence. If 7 was in position 4, it is now in position 10.

I hope you enjoy the program. You might like to improve it by adding letters in the sequence. Have fun!

Neville Predebon
West Preston Vic

```
5 REM - Shuffle -
6 REM - N. Predebon, 1983
7 CLS
8 SEC=VAL(RIGHT$(TIME$,2))
9 FOR SPUR=1 TO SEC: INIT=RND(1): NEXT SPUR
10 MOVE=1: A$=""
15 GOSUB 260
20 FOR NUM=0 TO 9
30 L=INT(RND(1)*10)+48
40 0=1
50 IF MID$(A$,0,1)=CHR$(L) THEN 30
60 IF 0 NUM THEN 0=0+1: GOTO 50
70 A$=A$+CHR$(L)
80 NEXT NUM
90 SOUND 1415,5: SOUND 1523,3: SOUND 1415,5
100 PRINT @ 127,"Move number";MOVE;": ":: PRINT A$
110 PRINT @ 209,"Reverse from number";: INPUT REV
120 IF REV<1 OR REV>9 THEN 110
130 B$=""
140 FOR SHUFFLE=10 TO REV STEP -1
150 B$=B$+MID$(A$,SHUFFLE,1)
160 NEXT SHUFFLE
170 A$=LEFT$(A$,REV-1)+B$
180 IF A$="0123456789" THEN 200
190 MOVE=MOVE+1: GOTO 90
200 CLS
210 PRINT @ 55,A$
220 PRINT @ 134,"YOU DID IT!!"
230 PRINT: PRINT TAB(9);"It took you";MOVE;"moves!!"
240 FOR FIN=0 TO 15: SOUND 1660,3: SOUND 1975,3: NEXT FIN
250 FOR DELAY=0 TO 99: NEXT DELAY: RUN
260 PRINT: FOR TITLE=1 TO 8: SOUND INT(RND(1)*200)+1100,5
270 PRINT "shuff";: FOR DELAY=0 TO 29: NEXT DELAY: NEXT TITLE
280 FOR DELAY=0 TO 99: NEXT DELAY
290 PRINT: PRINT "This is a game of logic. You have to try";
300 PRINT "to get the numbers in their proper order";
310 PRINT TAB(2);"by 'reversing' part of the sequence."
320 PRINT TAB(2);"Don't rush it - you've got forever!!"
330 PRINT @ 290,"hit any key to begin";
340 IF INKEY$="" THEN 340 ELSE CLS: RETURN
```

THE SCRAMBLER

```
CLOAD PRG1 <THE SCRAMBLER>
PRINT PEEK(16633)
IF >= 2 THEN POKE16548,PEEK(16633)-2:POKE16549,PEEK(16634)
IF = 0 OR 1 THEN POKE16548,PEEK(16633)+254:POKE16549,PEEK(16634)-1

CLOAD PRG2 <YOUR PROGRAM>
POKE16548,233:POKE16549,66
```

An unbreakable, triple A security system for the Level 2 TRS-80 and System 80 computers.

Scramble your programs so that when they are loaded they cannot be executed without the correct passcode.

Firstly, type in or append your program to LISTING 1, beginning it at line 10. When you are ready to save your program, type RUN and enter three security numbers. THE SCRAMBLER will go to work, encrypting everything beyond line 9. This can then be CSAVEd without any fear of someone else running or listing it.

When you decide to work on your program next, load and run it and then enter the same three numbers. Your program will be decoded back to its original state.

There are 125,000 different combinations so it is highly unlikely that anyone who doesn't know your numbers can crack it. But, don't forget the numbers since if even one of the numbers is slightly out, the decoding will make your program even more obscure. Line 0: This is free so you can enter – 0 GOTO 10. Typing this will make sure your program is not encoded while you are debugging your program. To scramble your program, just delete 0.

Robert Young
Thornlie WA

```
1 CLS:?"THE▲SCRAMBLER▲SECURITY▲SYSTEM":?
"WRITTEN▲5-1-83▲BY▲ROBERT▲E.▲YOUNG":?"VE
RSION▲1.6":?:?

2 INPUT"ENTER▲FIRST▲SECURITY▲NUMBER▲(FRO
M▲1▲TO▲200)";A:INPUT"ENTER▲SECOND▲SECURI
TY▲NUMBER▲(FROM▲1▲TO▲25)";B:INPUT"ENTER▲
THIRD▲SECURITY▲NUMBER▲(FROM▲1▲TO▲25)";C

3 FORD=17694TOPEEK(16633)+256*PEEK(16634
):IFPEEK(D)=0,D=D+5

4 E=PEEK(D)+A-255

5 IFF/B=INT(F/B),E=E+B

6 IFF/C=INT(F/C),E=E+C

7 IFE<0,E=E+255

8 IFE>255,E=E-255

9 ?F:POKED,E:F=F+1:NEXT:IFPEEK(17433)=20
6,POKE17433,205:POKE17435,206:POKE17459,
205:POKE17481,205:?"DECODING▲COMPLETE":E
ND:ELSEPOKE17433,206:POKE17435,205:POKE1
7459,206:POKE17481,206:?"ENCODING▲COMPLE
TE":END

10 REM  $ APPEND OR WRITE PROGRAM HERE $

11 REM  *

12 REM  *

13 REM  *

14 REM  *

15 REM  *
```

LISTING 1

NOTE - ▲ REFERS TO A SPACE. THIS PROGRAM MUST BE TYPED EXACTLY AS IT IS SET OUT IN ORDER FOR IT TO WORK CORRECTLY!!!

WAGES, SALARIES, TAX & APPORTIONMENTS

This program can be adapted to any small business where staff and materials are involved. Names of employees and hourly rates of pay can be easily changed. If there are less than ten employees enter zeros or merely press the ENTER key. If there are more than ten, then alter the appropriate lines i.e. 2000 FOR N1 to 10 etc. Names, amounts and descriptions can be altered at will provided the rules of syntax are observed. To run the program as it stands it is only necessary to enter the number of hours worked in each category as the computer asks for them, enter zero or pass. However in the apportionments a careful assessment of percentages should be made and the total must be 100 per cent. Where no work is carried out a zero must be entered.

Upon running this program you will be asked two questions: "what is the total amount allocated for this project?" and;

```
800 CLS
805 REM              .....BY ROBERT IAN BAALTE.....
807 REM                        '382
810 PRINT"   ... WAGES SALARIES TAX AND APPORTIONMENTS ...    "
820 PRINT"-----------------------------------------------"
830 PRINT
840 PRINT"THIS PROGRAMME CAN BE ADAPTED TO SUIT ANY FORM OF  "
850 PRINT"SMALL BUSINESS ACCOUNTING WHERE STAFF IS EMPLOYED"
860 PRINT"AND MATERIALS USED.                               "
892 PRINT
894 PRINT
896 PRINT"    THE FOLLOWING IS AN EXAMPLE          "
900 FOR Q = 1 TO 5000 : NEXT
920 CLS : CLEAR 2000
940 D$="$$#####.####.##"  NO$="NOTES :"  CO$="COINS :"
960 DIM AA(12),CU(13),BB$(12)
980 FOR N=1 TO 12 : READ BB$(N) : NEXT
1000 DATA $50 =,$20 =,$10 = $5 = $2 =,$1 =,50 CENT =,20 CENT =
10 CENT =,5 CENT =,2 CENT =,1 CENT =
1020 FOR N=1 TO 7 : READ AP$(N) : NEXT
1040 DATA 1. CLEARING,2. EARTHWORKS,3. DRAINAGE,4. CULVERTS,5.
FORMATION,6. SURFACING,7. RESERVE FUND
1060 PRINT:PRINTCHR$(23)".........SHIRE OF NOOSA.........":PRINT
1080 PRINT"            WORKSHEET":PRINT
1100 PRINT"      PROJECT NUMBER 17/83":PRINT
1120 PRINT"    SPUR ROAD, LOCATION 4/46"
1140 FOR Q = 1 TO 4000 : NEXT
1160 CLS:PRINT:PRINT:INPUT"ENTER TOTAL CONFIRMED COST OF PROJECT";TB#(1)
1180 PRINT:INPUT"ENTER PERCENTAGE ESTIMATED FOR LABOUR";P
1200 TB#(2)=TB#(1)*P/100 : BA#(7)=TB#(2)*1.25/100
1220 TB#(3)=TB#(1)-TB#(2) : TB#(2)=TB#(2)-BA#(7)
1240 CLS:PRINT:PRINT"YOUR LABOUR BUDGET, LESS RESERVE, IS :"
:PRINTUSING D$;TB#(2
1260 PRINT"ENTER THE PERCENTAGE ESTIMATED FOR EACH CATEGORY :":PRINT
1280 FOR N=1 TO 6 : PRINT AP$(N) : NEXT :PRINT
1300 PC=0 : FOR N=1 TO 6
1320 PRINT@704,"PROGRESSIVE %AGE = ";PC
1340 PRINT AP$(N) : INPUT W : PC(N)=W : BA#(N)=TB#(2)*W/100
1360 PC = PC+W : NEXT
1380 CLS: PRINT"YOUR BUDGET HAS BEEN ASSESSED AS FOLLOWS :":PRINT
1400 PRINT,"TOTAL COST :";:PRINT USING D$;TB#(1)
1420 PRINT,"MATERIALS :";:PRINT USING D$;TB#(3)
1440 PRINT"TOTAL LABOUR :";:PRINT USING D$;TB#(2)
1460 FOR N=1 TO 6 : PRINT AP$(N);:PRINT USING D$;BA#(N);
:PRINT,PC(N);"%" : NEXT
1480 PRINT,AP$(7);:PRINT USING D$;BA#(7):PRINT," -"
1500 PRINT@832,"PLEASE CHECK. TO ALTER, PRESS A.
CONTINUE WITH ANY OTHER KEY."
1520 E$=INKEY$ : IF E$="" THEN 1520 ELSE IF E$=CHR$(65)
THEN 1180 ELSE 2000
2000 FOR N=1 TO 10 : READ N$(N) : NEXT
2020 DATA 1.FISHER JOHN,2.ARTHUR JAMES,3.IVES SIMON,4.
STEED PAUL,5.DAY TERRENCE,
6.WILLIAMS A.B.,7.ARKWRIGHT J.,8.JOHNSON S.,9.
AMOS CARL,10.TREDOS ERICA
2040 FOR N = 1 TO 10 : READ C$(N),R(N) : NEXT
2060 DATA LABOURER,5.25,QUARRYMAN,6.50,HAMMER & DRILL,6.75,
SCOOPMAN,6.65,POWDER
MONKEY,7.50,TRUCK DRIVER,8.00,GRADER DRIVER,8.25,CARPENTER,7.15
SECRETARY,7.00,F
OREMAN,9.25
2080 GOSUB 10000
2100 CLS : PRINT"ENTER HOURS WORKED IN EACH CATEGORY :"
2120 DIM H(10,10),W(10,10) : TG=0 : FOR N = 1 TO 10:LC(N)=0:NEXT
2140 FOR N=1 TO 10
2160 PRINT PRINT"HOURS WORKED BY ";N$(N),"  "
2180 FOR M = 1 TO 10
2200 PRINT C$(M);" "
2220 INPUT H(N,M) : NEXT
3000 GW=0 : FOR X = 1 TO 10
3020 W = H(N,X)*R(X) : W=INT(W*100+.5)/100 :
    W(N,X)=W : LC(X)=LC(X)+W : GW=GW+W
: NEXT
3040 GW(N)=GW   TG=TG+GW(N)   TG = INT(TG*100+.5)/100  NEXT N
3050 ZZ=0 : FOR M=1 TO 12 : AA(M)=0 : NEXT
3060 TT=0 : NT=0 : FOR N=1 TO 10
3080 IF GW(N) =< 200 THEN T=0 ELSE T=32
3100 IF GW(N) => 800 THEN T=45 ELSE IF GW(N) => 1200 THEN T = 64
3120 T(N)=GW(N)*T/100 : T(N)=INT(T(N)*100+.5)/100 : TT=TT+T(N)
3140 NW(N)=GW(N)-T(N) : GOSUB 9200
3160 NT = NT+NW(N) : NEXT
4000 CLS : PRINT,"APPORTIONMENTS"
4020 PRINT"PLEASE INSERT APPROPRIATE PERCENTAGES BELOW :-"
4030 PRINT"CURRENT LABOUR COST (ALL CATEGORIES) = $";TG
4040 PRINT: PRINT,"   CATEGORY :","% BUDGETED :"
4045 FOR N=1 TO 6 : PRINT,AP$(N),"      ";PC(N) : NEXT
4080 FOR N = 1 TO 6 : AE(N) = 0 : NEXT
4100 Q = 0
4120 FOR Z = 1 TO 6
4140 PRINT@ 704,"PROGRESSIVE %AGE = ";Q
4160 PRINTAP$(Z); : INPUT R : C(Z)=TG*R/100 : AE(Z)=AE(Z)+C(Z)
4180 Q=Q+R : NEXT
4200 CLS : PRINT@ 538, "PLEASE WAIT !"
4210 FOR Q = 1 TO 4000 : NEXT
4220 CLS : PRINT "DATA PROCESSING HAS BEEN COMPLETED."
4240 PRINT"INFORMATION IS AVAILABLE AS FOLLOWS :"
4260 PRINT:PRINT,"1. HOURS WORKED AND VALUES"
4280 PRINT,"2. WAGES AND TAXATION"
4300 PRINT,"3. ACCUMULATED LABOUR COSTS"
4320 PRINT,"4. CURRENCY TABLE"
4340 PRINT,"5. EXPENDITURE"
4360 PRINT,"6. ORIGINAL BUDGET"
4380 PRINT,"7. ARRAY STORAGE"
4400 PRINT : PRINT"PLEASE PRESS THE REQUIRED NUMBER."
4420 E$=INKEY$ : IF E$="" THEN4420
4440 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
5000 CLS : PRINT@77 ,"PLEASE ENTER EMPLOYEE'S NUMBER :"
5020 INPUT EN
5040 CLS : PRINT N$(EN);" :"
5060 PRINT,"CATEGORY","HOURS WORKED"," VALUE ($GROSS)"
5080 FOR X = 1 TO 10
5100 PRINT,C$(X),H(EN,X),
5120 PRINTUSING D$;W(EN,X) : NEXT
5130 PRINT,"TOTAL :";:PRINT USING D$;GW(EN)
5140 PRINT : GOSUB 11000
5160 E$=INKEY$ : IF E$="" THEN 5160
5180 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
5500 CLS : PRINT TAB(18)"WAGES AND TAXATION"
5520 PRINT,"   GROSS WAGE","TAX PAYABLE","NET WAGE"
5540 FOR N=1 TO 10
5560 PRINTN$(N);:PRINT USING D$;GW(N),T(N),NW(N) : NEXT
5580 PRINT"TOTALS :";: PRINT USING D$;TG,TT,NT : PRINT
5600 GOSUB 11000
5620 E$=INKEY$ : IF E$="" THEN 5620
5640 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
6000 CLS : PRINTTAB(22)"CURRENT LABOUR COSTS"
6020 PRINT"CATEGORY","  HOURLY RATE","COST TO DATE"
6040 FOR N = 1 TO 10
6060 PRINT C$(N);:PRINT USING D$;R(N);:PRINT  USING D$;LC(N) : NEXT
6080 PRINT"TOTAL :";:PRINT USING D$;TG
6100 GOSUB 11000
6120 E$=INKEY$ : IF E$="" THEN 6120
6140 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
6500 CLS : PRINT "CURRENCY TABLE"
6520 PRINT "TOTAL PAYOUT = $";ZZ
6540 PRINT : PRINT NO$, CO$
6560 FOR M=1 TO 5 : PRINT BB$(M),BB(M),BB$(M+6),BB(M+6) : NEXT
6580 K=BA(1)*50+BA(2)*20+BA(3)*10+BA(4)*5+BA(5)*2+BA(6)
6600 KK=BA(7)*50+BA(8)*20+BA(9)*10+BA(10)*5+BA(11)*2+BA(12)
KK=KK/100 : KK=INT
(KK*100+.5)/100
6620 PRINT "$" K "$" KK
6640 GOSUB 11000
```

```
6660 E$=INKEY$ : IF E$ = "" THEN 6660
6680 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
7000 AE(0) = 0 : FOR N = 1 TO 6 : AE(0) = AE(0)+AE(N) : NEXT
7020 CLS : PRINTTAB(26)"EXPENDITURE."
7040 PRINT:PRINT"CATEGORY","   BUDGET","   ACTUAL","   BALANCE"
7060 FOR N=1 TO 7
7080 PRINTAP$(N);:PRINT,USINGD$;BA#(N);:PRINT,USINGD$;PE(N);:PRINT,
USINGD$;BA#(N)-AE(N) : NEXT
7100 PRINT:PRINT"TOTALS :";:PRINT,USINGD$;TB#(2);:PRINT,USINGD$;
AE(0);:PRINT,USINGD$;TB#(2)-AE(0)
7120 PRINT"PRESS : E - TO INCREASE EXPENDITURE ; T - TO TRANSFER
7140 GOSUB 11000                                          FUNDS."
7160 E$=INKEY$ : IF E$="" THEN 7160
7180 IF E$=CHR$(84) THEN7300 ELSE IF E$=CHR$(69) THEN 7200
7190 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
7200 CLS : PRINT"ENTER ADDITIONAL COSTS IN EACH CATEGORY :"
7220 FOR N=1 TO 7 : PRINT AP$(N) : INPUT E(N)
7240 AE(N)=AE(N)+E(N) : AE(0)=AE(0)+E(N)
7260 NEXT : GOTO 7020
7300 CLS : PRINT,"TRANSFER TABLE"
7320 PRINT,"CATEGORY","   EXPENDITURE","   BALANCE"
7340 FOR N=1 TO 7
7360 PRINT,AP$(N);:PRINT,USINGD$;AE(N);:PRINT,USINGD$;BA#(N)-AE(N) :
7380 PRINT:INPUT"ENTER NUMBERS  'FROM'  AND  'TO'";X,Z          NEXT
7400 INPUT"ENTER AMOUNT TRANSFERABLE";Y
7410 IF X = 7 THEN AE(X) = AE(X)+Y ELSE BA#(X) = BA#(X)-Y
7420 BA#(Z) = BA#(Z)+Y
7440 GOTO 7020
7500 CLS : PRINTTAB(22)"TOTAL BUDGET  (COPY)."
7520 PRINT:PRINT"YOUR BUDGET ASSESSMENT WAS AS FOLLOWS :"
7540 PRINT,"TOTAL PROJECT COST :";:PRINT,USINGD$;TB#(1)
7560 PRINT,"MATERIALS          :";:PRINT,USINGD$;TB#(3)
7580 PRINT"TOTAL LABOUR :";:PRINT,USINGD$;TB#(2)
7600 FOR N=1 TO 6:PRINT,AP$(N);:PRINT,USING D$;TB#(2)*PC(N)/100);:
PRINT,PC(N);" %":NEXT
7620 PRINT,AP$(7);:PRINT,USINGD$;BA#(7);:PRINT,"  -"
7640 GOSUB 11000
7660 E$=INKEY$ : IF E$="" THEN 7660
7680 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
8000 CLS : PRINT"...DATA IS STORED WITHIN THE FOLLOWING ARRAYS..."
8020 PRINT:PRINT"APPORTIONMENTS  :  AP$(N)    BUDGET      :     BA(N)"
8040   PRINT"JOB CATEGORIES   :   C$(N)    ACTUAL EXP.   :     PE(N)"
8060   PRINT"HOURS WORKED     :  H(N,M)    GROSS WAGE    :     GW(N)"
8080   PRINT"PAY RATES/HR.    :    R(N)    NET WAGE      :     NW(N)"
8100   PRINT"STAFF NAMES      :   N$(N)    TAXATION      :      T(N)"
8120   PRINT"LABOUR COSTS     :   LC(N)    TOTAL EXP.    :     PE(0)"
8140   PRINT"PROJECT COST     :  TB#(1)    TOTAL LABOUR  :    TB#(2)"
8160   PRINT"MATERIAL COST    :  TB#(3)    BUDGET %AGES  :     PC(N)"
8180 GOSUB 11000
8200 E$=INKEY$ : IF E$="" THEN 8200
8220 ON VAL(E$) GOTO 5000,5500,6000,6500,7000,7500,8000
9000 END
9200 CU(0) = FIX(NW(N)) : CU(13) = FIX((NW(N)-CU(0))*100+.01) :
9220 FOR M=1 TO 12 : CU(M)=0 : NEXT                       ZZ=ZZ+NW(N)
9240 CU(1)=FIX(CU(0)/50) : R=CU(0)-CU(1)*50
9260 IF R=>20 THEN CU(2)=FIX(R/20) : R=R-CU(2)*20
9280 IF R=>10 THEN CU(3)=FIX(R/10) : R=R-CU(3)*10
9300 IF R=>5  THEN CU(4)=FIX(R/5)  : R=R-CU(4)*5
9320 IF R=>2  THEN CU(5)=FIX(R/2)  : R=R-CU(5)*2
9340 CU(6) = R
9360 IF CU(13)=>50 THEN CU(7)=FIX(CU(13)/50) : CU(13)=CU(13)-CU(7)*50
9380 Q=CU(13) : IF Q=>20 THEN CU(8)=FIX(Q/20) : Q=Q-CU(8)*20
9400 IF Q=>10 THEN CU(9)=FIX(Q/10) : Q=Q-CU(9)*10
9420 IF Q=> 5 THEN CU(10)=FIX(Q/5) : Q=Q-CU(10)*5
9440 IF Q=> 2 THEN CU(11)=FIX(Q/2) : Q=Q-CU(11)*2
9460 CU(12) = Q
9480 FOR M=1 TO 12 : ??(M) = ??(M)+CU(M) : NEXT
9500 RETURN
10000 CLS:PRINT,"PLEASE VERIFY HOURLY RATES :" : PRINT
10100 FOR N = 1 TO 10
10200 PRINT C$(N);:PRINT,USING D$;R(N) : NEXT : PRINT
10300 PRINT"IF CORRECT, PRESS ANY KEY, ELSE RE-ENTER(EDIT 9060)
10400 PRINT"IF NECESSARY, ALTER TAX SCALES (EDIT 9000-9100)"
10450 E$ = INKEY$ : IF E$ = "" THEN 10450 ELSE RETURN
10500 END
11000 PRINT@ 896,"1.HRS/VALUE","2.WAGES/TAX","3.LAB.COSTS"
11100 PRINT"4.CURRENCY","5.EXPENDITURE","6.BUDGET","7.ARRAYS";
11200 RETURN
11300 END
```

WAGES AND TAXATION

	GROSS WAGE	TAX PAYABLE	NET WAGE
1.FISHER JOHN	$420.00	$134.40	$285.60
2.ARTHUR JAMES	$460.00	$147.20	$312.80
3.IVES SIMON	$420.00	$134.40	$285.60
4.STEED PAUL	$452.50	$144.80	$307.70
5.DAY TERRENCE	$526.00	$168.32	$357.68
6.WILLIAMS A.B.	$524.80	$167.94	$356.86
7.ARKWRIGHT J.	$465.60	$148.99	$316.61
8.JOHNSON S.	$648.00	$207.36	$440.64
9.AMOS CARL	$756.00	$241.92	$514.08
10.TREDOS ERICA	$560.00	$179.20	$380.80
TOTALS :	$5,232.90	$1,674.53	$3,558.37

EXPENDITURE.

CATEGORY	BUDGET	ACTUAL	BALANCE
1. CLEARING	$1,777.50	$523.29	$1,254.21
2. EARTHWORKS	$6,221.25	$1,831.52	$4,389.73
3. DRAINAGE	$888.75	$261.65	$627.11
4. CULVERTS	$2,133.00	$627.95	$1,505.05
5. FORMATION	$1,422.00	$418.63	$1,003.37
6. SURFACING	$5,332.50	$1,569.87	$3,762.63
7. RESERVE FUND	$225.00	$0.00	$225.00
TOTALS :	$17,775.00	$5,232.90	$12,542.10

CURRENCY TABLE
TOTAL PAYOUT = $ 3558.37

NOTES :		COINS :	
$50 =	67	50 CENT =	9
$20 =	5	20 CENT =	4
$10 =	6	10 CENT =	8
$5 =	6	5 CENT =	3
$2 =	5	2 CENT =	4
$1 =	3	1 CENT =	4
	$ 3558		$ 6.37

"what percentage do you wish to allow for labour?". Materials are not considered in this program. Allowance for this should be made when setting the percentage for labour.

Here is a test submission for the Wages or Salaries, Tax and Apportionments program which produced the printed tables.
Total cost of the Project (including materials) $25,000 Percentage for labour 72 per cent.
Hours, Categories, Wages, are produced from this in-put.
1. Fisher John. Labourer 80 hours
2. Arthur James. Labourer 48, Quarryman 32 hours
3. Ives Simon. Labourer 80 hours
4. Steed Paul. Labourer 54 Quarryman 26 hours
5. Day Terrence. Quarryman 56 Hammer & Drill 24 hours
6. Williams A.B. Quarryman 56 Hammer & Drill 12 Scoopman 12 hours
7. Arkwright J. Labourer 56 Carpenter 24 hours
8. Johnson S. Truck-driver 48 Grader-driver 32 hours
9. Amos Carl. Powder-monkey 12 Foreman 72 hours
10. Tredos Erica. Secretary 80 hours
Percentages allotted to each section (in both instances)
1. Clearing 10%
2. Earthworks 35%
3. Drainage 5%
4. Culverts 12%
5. Formation 8%
6. Surfacing 30%
Total labour cost $17,775.00
materials $ 7,000.00 reserve $ 225.00
Note: There is rather a long pause after entering the Employee's hours of work. This is due to the number of calculations to be worked out at this stage. In the third repeat of the program on the cassette this has been taken care of with a suitable explanation which is not on the print-out.

Robert van Raalte
Nedlands WA

POKES FOR YOUR EYES

This program divides the screen into eight, numbering each Area; 1A, 1B, etc. Each Area can be called from a Menu, and the memory addresses for that area are printed to the screen. Thus each one-eighth of the screen can be 'exploded' to fill the screen with the memory addresses for that Area.

I've called it 'POKES FOR YOUR EYES', because that is what it does: puts the POKE addresses, in full, before your eyes.

Arthur Pittard
Fairfield NSW

```
100 CLS:PRINT@ 140,"******** DEVELOPED BY ********"
110 PRINT@210,"ARTHUR PITTARD
                36 HUBERT STREET,FAIRFIELD,2165
                      N.S.W. AUSTRALIA
                      ******************"
120 PRINT@523,"POKES  -   FOR  -   YOUR  -   EYES

                TRS-80 MODEL III  BASIC LEVEL II

                  THE SCREEN 'LAYOUT' FOLLOWS"
130 CLEAR 500
140 PRINT@790,"PRESS ANY ALPHABET KEY TO CONTINUE"
150 0$=INKEY$:IF 0$="" THEN 150 ELSE 160
160 CLS:FOR OVER=0 TO 63:SET(OVER,0):NEXT
170 FOR OVER=0 TO 63:SET(OVER,11):NEXT
180 FOR DOWN=0 TO 11:SET(0,DOWN):NEXT
190 FOR DOWN=0 TO 11:SET(63,DOWN):NEXT
200 PRINT@135,"THIS IS AREA '1 A'";
210 FOR OVER=64TO127:SET(OVER,12):NEXT
220 FOR OVER=64TO127:SET(OVER,23):NEXT
230 FOR DOWN=12TO 23:SET(64,DOWN):NEXT
240 FOR DOWN=12 TO 23:SET(127,DOWN):NEXT
250 PRINT@422,"THIS IS AREA '2 B'";
260 FOR OVER=0TO63:SET(OVER,24):NEXT
270 FOR OVER=0TO63:SET(OVER,35):NEXT
280 FOR DOWN=24TO35:SET(0,DOWN):NEXT
290 FOR DOWN=24TO35:SET(63,DOWN):NEXT
300 PRINT@646,"THIS IS AREA '3 A'";
310 FOR OVER =64 TO 127:SET(OVER,36):NEXT
320 FOR OVER =64 TO 127:SET(OVER,47):NEXT
330 FOR DOWN=36 TO 47:SET(64,DOWN):NEXT
340 FOR DOWN=36 TO 47:SET(127,DOWN):NEXT
350 PRINT@870,"THIS IS AREA '4 B'";
360 FOR OVER= 64 TO 127:SET(OVER,0):NEXT
370 FOR OVER= 64TO127:SET(OVER,11):NEXT
380 FOR DOWN= 0 TO 11:SET(64 ,DOWN):NEXT
390 FOR DOWN=0  TO 11:SET(127,DOWN):NEXT
400 PRINT@166,"THIS IS AREA '1 B'";
410 FOR OVER=0 TO 63:SET(OVER,12):NEXT
420 FOR OVER=0 TO63:SET(OVER,23):NEXT
430 FOR DOWN=12 TO23:SET(0,DOWN):NEXT
440 FOR DOWN=12 TO 23:SET(63,DOWN):NEXT
450 PRINT@390,"THIS IS AREA '2 A'";
460 FOR OVER=64TO127:SET(OVER,24):NEXT
470 FOR OVER=64 TO127:SET(OVER,35):NEXT
480 FOR DOWN=24 TO35:SET(64,DOWN):NEXT
490 FOR DOWN=24 TO 35:SET(127,DOWN):NEXT
500 PRINT@678,"THIS IS AREA '3 B'";
510 FOR OVER=0 TO 63:SET(OVER,36):NEXT
520 FOR OVER=0 TO 63:SET(OVER,47):NEXT
530 FOR DOWN=36 TO 47:SET(0,DOWN):NEXT
540 FOR DOWN =36 TO 47:SET(63,DOWN):NEXT
550 PRINT@838,"THIS IS AREA '4 A'";
560 PRINT@898,"* * * * PRESS ANY  KEY * * *";@932,"TO CONTINUE * * * *";
570 L$=INKEY$:IF L$=""570
580 CLS:PRINT@12,"THE SCREEN IS DIVIDED INTO 8 AREAS.
              SELECT THE AREA YOU REQUIRE FROM THE MENU.
                PRESS THE CORRESPONDING NUMBER & <ENTER>."
590 PRINT@204,"PRESS ANY ALPHABET KEY TO RETURN TO THE MENU"
600 PRINT@334,"MENU":PRINT@462,
             "1 - AREA 1 A            5 - AREA 1 B
              2 - AREA 2 A            6 - AREA 2 B
              3 - AREA 3 A            7 - AREA 3 B
              4 - AREA 4 A            8 - AREA 4 B"
610 PRINT"PRESS DESIRED NUMBER KEY  -THEN PRESS <ENTER>"
620 INPUTZ:IF Z>8 THEN 600
630 DEFSTRA-M
640 A="1 1 1 1 1 1 1 1 1 1 ":B="2 2 2 2 2 2 2 2 2 2 "
650 C="3 3 3 3 3 3 3 3 3 3 ": D="4 4 4 4 4 4 4 4 4 4 "
660 E="5 5 5 5 5 5 5 5 5 5 ": F="6 6 6 6 6 6 6 6 6 6 "
670 G="7 7 7 7 7 7 7 7 7 7 ": H="8 8 8 8 8 8 8 8 8 8 "
680 I="9 9 9 9 9 9 9 9 9 9 ": J="0 1 2 3 4 5 6 7 8 9 "
690 AA="0 0 0 0 0 0 0 0 0 0 "
700 K="15000 +":L="16000 +":M="ROW #":GOTO720
710 REM
```

```
720 ON Z GOTO 730,940,1150,1360,1570,1780,1990,2200
730 CLS:PRINT@15,K;@39,M+" 1 A"
740 PRINT@64,C+C+C+LEFT$(C,4)
750 PRINT@128,F+G+H+LEFT$(I,4)
760 PRINT@192,J+J+J+LEFT$(J,3)
770 REM
780 PRINT@271,K;@295,M+" 2 A"
790 PRINT@320,D+D+D+LEFT$(D,4)
800 PRINT@384,RIGHT$(B,12)+C+D+LEFT$(E,12)
810 PRINT@447,RIGHT$(J,13)+J+J+LEFT$(J,11)
820 REM
830 PRINT@527,K;@551,M+" 3 A"
840 PRINT@576,LEFT$(D,4)+D+E+E
850 PRINT@640,RIGHT$(H,4)+I+AA+LEFT$(A,19)
860 PRINT@704,RIGHT$(J,4)+J+J+J
870 REM
880 PRINT@783,K;@807,M+" 4 A"
890 PRINT@832,E+E+E+RIGHT$(E,4)
900 PRINT@896,LEFT$(E,16)+F+G+LEFT$(H,7)
910 PRINT@960,RIGHT$(J,16)+J+J+LEFT$(J,7);
920 GOTO 2400
930 REM
940 CLS:PRINT@15,K;@39,M+" 5 A"
950 PRINT@64,F+F+F+LEFT$(F,3)
960 PRINT@128,RIGHT$(A,8)+B+C+LEFT$(D,15)
970 PRINT@192,RIGHT$(J,8)+J+J+LEFT$(J,15)
980 REM
990 PRINT@271,K;@295,M+" 6 A"
1000 PRINT@320,F+F+G+LEFT$(G,4)
1010 PRINT@384,H+I+AA+LEFT$(A,3)
1020 PRINT@448,J+J+J+LEFT$(J,3)
1030 REM
1040 PRINT@527,K;@551,M+" 7 A"
1050 PRINT@576,G+G+G+LEFT$(G,3)
1060 PRINT@640,RIGHT$(D,12)+E+F+LEFT$(G,11)
1070 PRINT@704,RIGHT$(J$,12)+J+J+LEFT$(J,12)
1080 REM
1090 PRINT@783,K;@807,M+" 8 A"
1100 PRINT@832,H+H+H+LEFT$(H,4)
1110 PRINT@896,LEFT$(AA,4)+A+B+LEFT$(C,19)
1120 PRINT@960,RIGHT$(J,4)+J+J+LEFT$(J,19);
1130 GOTO 2400
1140 REM
1150 CLS:PRINT@15,K;@39,M+" 9 A"
1160 PRINT@64,H+H+LEFT$(H,16)LEFT$(I,7)
1170 PRINT@128,LEFT$(G,16)+H+I+LEFT$(AA,8)
1180 PRINT@192,RIGHT$(J,16)+J+J+LEFT$(J,7)
1190 REM
1200 PRINT@271,K;@295,M+"10 A"
1210 PRINT@320,I+I+I+LEFT$(I,4)
1220 PRINT@384,LEFT$(C,8)+D+E+LEFT$(F,15)
1230 PRINT@448,RIGHT$(J,8)+J+J+LEFT$(J,15)
1240 REM
1250 PRINT@518,"* * ";L;"  * *";@548,M+"11 A"
1260 PRINT@576,AA+AA+AA+LEFT$(AA,4)
1270 PRINT@640,AA+A+B+LEFT$(C,3)
1280 PRINT@704,J+J+J+LEFT$(J,3)
1290 REM
1300 PRINT@774,"* * ";L;"  * * ";@804,M+"12 A"
1310 PRINT@832,AA+AA+AA+LEFT$(AA,4)
1320 PRINT@896,LEFT$(F,12)+G+H+LEFT$(I,11)
1330 PRINT@960,RIGHT$(J,12)+J+J+LEFT$(J,11);
1340 GOTO2400
1350 REM
1360 CLS:PRINT@ 9,"* * ";L;"  * * ";@34,M+" 13 A"
1370 PRINT@64,A+A+A+LEFT$(A,3)
1380 PRINT@128,LEFT$(B,4)+C+D+E
1390 PRINT@192,RIGHT$(J,4)+J+J+J
1400 REM
1410 PRINT@265,"* *   ";L;"  * * ";@290,M+"14 A"
1420 PRINT@320,LEFT$(A,16)+B+B+LEFT$(B,8)
1430 PRINT@384,LEFT$(I,16)+AA+A+LEFT$(B,8)
1440 PRINT@448,RIGHT$(J,16)+J+J+LEFT$(J,8)
1450 REM
1460 PRINT@521,"* * ";L;"  * *";@546,M+"15 A"
1470 PRINT@576,B+B+B+LEFT$(B,4)
1480 PRINT@640,RIGHT$(E,8)+F+G+LEFT$(H,15)
1490 PRINT@704,RIGHT$(J,8)+J+J+LEFT$(J,15)
1500 REM
1510 PRINT@777,"* *   ";L;"  * *";@802,M+"16 A"
1520 PRINT@832,C+C+C+LEFT$(C,3)
1530 PRINT@896,B+C+D+LEFT$(E,3)
1540 PRINT@960,J+J+J+LEFT$(J,3);
1550 GOTO2400
1560 REM
1570 CLS:PRINT@15,K;@39,M+" 1 B"
1580 PRINT@64,LEFT$(C,16)+D+D+LEFT$(D,8)
1590 PRINT@128,LEFT$(I,16)+AA+A+LEFT$(B,8)
1600 PRINT@192,RIGHT$(J,16)+J+J+LEFT$(J,7)
1610 REM
1620 PRINT@271,K;@295,M+" 2 B"
1630 PRINT@320,D+D+D+LEFT$(D,4)
1640 PRINT@384,LEFT$(E,8)+F+G+LEFT$(H,16)
1650 PRINT@448,RIGHT$(J,8)+J+J+LEFT$(J,16)
1660 REM
1670 PRINT@527,K;@551,M+" 3 B"
1680 PRINT@576,E+E+E+LEFT$(E,3)
1690 PRINT@640,B+C+D+LEFT$(E,4)
1700 PRINT@704,J+J+J+LEFT$(J,3)
1710 REM
1720 PRINT@783,K;@807,M+" 4 B"
1730 PRINT@832,LEFT$(E,12)+E+F+LEFT$(F,12)
1740 PRINT@896,LEFT$(H,12)+I+AA+LEFT$(A,11)
1750 PRINT@960,RIGHT$(J,12)+J+J+LEFT$(J,11);
1760 GOTO2400
1770 REM
1780 CLS:PRINT@15,K;@39,M+" 5 B"
1790 PRINT@64,F+F+F+LEFT$(F,4)
1800 PRINT@128,LEFT$(D,4)+E+F+G
1810 PRINT@192,RIGHT$(J,4)+J+J+LEFT$(J,20)
1820 REM
1830 PRINT@271,K;@295,M+" 6 B"
1840 PRINT@320,G+G+G+LEFT$(G,4)
1850 PRINT@384,LEFT$(A,16)+B+C+LEFT$(D,8)
1860 PRINT@448,RIGHT$(J,16)+J+J+LEFT$(J,8)
1870 REM
1880 PRINT@527,K;@551, M+" 7 B"
1890 PRINT@576,G+G+LEFT$(G,8)+LEFT$(H,16)
1900 PRINT@640,LEFT$(G,8)+H+I+LEFT$(AA,16)
1910 PRINT@704,RIGHT$(J,8)J+J+LEFT$(J,16)
1920 REM
1930 PRINT@783,K;@807,M+" 8 B"
1940 PRINT@832,H+H+H+LEFT$(H,3)
1950 PRINT@896,D+E+F+LEFT$(G,3)
1960 PRINT@960,J+J+J+LEFT$(J,3);
1970 GOTO2400
1980 REM
1990 CLS:PRINT@15,K;@39,M+" 9 B"
2000 PRINT@64,I+I+I+LEFT$(I,3)
2010 PRINT@128,LEFT$(AA,12)+A+B+LEFT$(C,12)
2020 PRINT@192,RIGHT$(J,12)+J+J+LEFT$(J,12)
2030 REM
2040 PRINT@271,K;@295,M+"10 B"
2050 PRINT@320,I+I+I+LEFT$(I,4)
2060 PRINT@384,LEFT$(F,4)+G+H+I
2070 PRINT@448,RIGHT$(J,4)+J+J+LEFT$(J,19)
2080 REM
2090 PRINT@518,"* *  ";L;"  * * ";@548,M+"11 B"
2100 PRINT@576,LEFT$(AA,4)+AA+AA+RIGHT$(AA,20)
2110 PRINT@640,LEFT$(C,16)+D+E+LEFT$(F,7)
2120 PRINT@704,RIGHT$(J,16)+J+J+LEFT$(J,7)
2130 REM
2140 PRINT@774,"* *  ";L;"  * * ";@804,M+"12 B"
2150 PRINT@832,LEFT$(AA,8)+A+A+LEFT$(A,16)
2160 PRINT@896,LEFT$(I,8)+AA+A+LEFT$(B,15)
2170 PRINT@960,RIGHT$(J,8)+J+J+LEFT$(J,15);
2180 GOTO2400
2190 REM
2200 CLS:PRINT@9,"* *  ";L;"  * *";@34,M+"13 B"
2210 PRINT@64,A+A+A+LEFT$(A,3)
2220 PRINT@128,F+G+H+LEFT$(I,4)
2230 PRINT@192,J+J+J+LEFT$(J,3)
2240 REM
2250 PRINT@265,"* *   ";L;"  * * ";@290,M+" 14 B"
2260 PRINT@320,B+B+B+LEFT$(B,4)
2270 PRINT@384,LEFT$(B,12)+C+D+LEFT$(E,12)
2280 PRINT@448,RIGHT$(J,12)+J+J+LEFT$(J,11)
2290 REM
2300 PRINT@521,"* *  ";L;"  * *";@546,M+" 15 B"
2310 PRINT@576,B+LEFT$(B,4)+C+C
2320 PRINT@640,LEFT$(H,4)+I+AA+A
2330 PRINT@704,RIGHT$(J,4)+J+J+LEFT$(J,19)
2340 REM
2350 PRINT@777,"* *   ";L;"  * * ";@802,M+"16 B"
2360 PRINT@832,C+C+C+LEFT$(C,4)
2370 PRINT@896,LEFT$(E,8)+F+G+LEFT$(H,15)
2380 PRINT@960,RIGHT$(J,16)+J+J+LEFT$(J,7);
2390 GOTO 2400
2400 P$=INKEY$:IF P$="" THEN 2400 ELSE 580
2500 REM
2600 REM
2700 REM AS MEMORY LOCATIONS 16 THOUSAND & OVER ARE IN THE
       MIDDLE, AS IT WERE, THEY HAVE BEEN IDENTIFIED
         WITH THE * * * & OFFSET TO ATTRACT ATTENTION
```

```
.....................................................................
.                                   ..                              .
.      THIS IS AREA '1 A'           ..      THIS IS AREA '1 B'      .
.....................................................................
.....................................................................
.                                   ..                              .
.      THIS IS AREA '2 A'           ..      THIS IS AREA '2 B'      .
.....................................................................
.....................................................................
.                                   ..                              .
.      THIS IS AREA '3 A'           ..      THIS IS AREA '3 B'      .
.....................................................................
.....................................................................
.      THIS IS AREA '4 A'           ..      THIS IS AREA '4 B'      .
. * * * * PRESS ANY  KEY * * * ..   TO CONTINUE * * * *            .
.....................................................................
```

See Note A below.

PRINT-OUT BELOW of AREA 1A (as seen on full screen)

```
                  15000 +                    ROW # 1 A
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 9 9
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
                  15000 +                    ROW # 2 A
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5
4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
                  15000 +                    ROW # 3 A
4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
8 8 9 9 9 9 9 9 9 9 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1
8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
                  15000 +                    ROW # 4 A
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
5 5 5 5 5 5 5 5 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 8 8 8
2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
```

A - This is LPVII's print-out. Actually the screen is filled (full) to the last line.

CLUEDO

User Machine: TRS-80 Level II - uses 14K. Designed for Mod 1 but will run on Mod III – 32K upwards but you must delete the Mem Size pokes in Line 1. Disk systems without DOS boosted. Operates with or without L/C installed.

Development: Originally the program was written entirely in Basic but the computer used to take about 40 seconds after each player's turn to scan possibilities and store and retrieve information from previous calls.

This was unacceptable so I delved into assembler for a while and the resultant machine code allows for very fast turnaround. If someone wants to disassemble it, it resides from 7273H to 74D3H.

Alan Goodison
Mooroopna Vic

```
1 POKE16561,114:POKE16562,114:GOSUB89
2 CLEAR500:DIMN$(21):DIMA$(6):DEFINTA-Z:Z=30000:Y=30150:Q=30160:GOSUB91:GOSUB92:
PRINT@770,CHR$(30):GOSUB57
3 PRINT@836,CHR$(31);:INPUT"Computer Player's number";N$:C1=VAL(N$):IFC1>NORC1<1
PRINT@836,"1 to "N" Please."CHR$(30):FORX=1TO1500:NEXT:GOTO3ELSEFORX=1TO21:POKEZ
+C1*21+X,7:NEXT
4 PRINT@900,CHR$(31);:INPUT"Who's going to deal";N$:D2=VAL(N$):IFD2>NORD2<1THENP
RINT@900,"1 to "N" Please."CHR$(30):FORX=1TO1500:NEXT:GOTO4ELSEGOSUB63:GOTO107
5 V=1:PRINT@640,"Enter your own cards (by number):";:PRINT@710,"Enter zero when
complete";:C3=1:GOTO58
6 CLS:W=0:PRINTTAB(3)"SUSPECTS";TAB(26)"IMPLEMENTS";TAB(47)"ROOMS":PRINT:PRINT"
"N$(1);TAB(24)N$(7);TAB(44)N$(13):PRINT" "N$(2);TAB(24)N$(8);TAB(44)N$(14):PRINT
" "N$(3);TAB(24)N$(9);TAB(44)N$(15):PRINT" "N$(4);TAB(23)N$(10);TAB(44)N$(16)
7 PRINT" "N$(5);TAB(23)N$(11);TAB(44)N$(17):PRINT" "N$(6);TAB(23)N$(12);TAB(44)N
$(18):PRINTTAB(44)N$(19):PRINTTAB(44)N$(20):PRINTTAB(44)N$(21):IFV=0THEN5ELSEPRI
NT@640,T#:PRINT:PRINT"Who's calling - Player no     (Zero for display - else 1 t
o"N")";
8 PRINT@793,;:INPUTU:PRINT@796,CHR$(30):IFU=0THEN79ELSET=U
9 INPUT"Suspect Person card  (1 to 6):  ";P:INPUT"Suspect Implement card  (7 to 1
2):  ";I:INPUT"Suspect Room card  (13 to 21):  ";S:IFP<1ORP>6ORI<7ORI>12ORS<13ORS>21
ORU<0ORU>NTHENT$="**  Try again  **":GOTO6ELSET$=""
10 CLS:A=T+1:IFA>NTHENA=1
11 PRINT:PRINTN$(P)"     "N$(I)"     "N$(S):PRINT:PRINT:IFA=C1THEN13ELSEPRINT"Ask
"A$(A):PRINT:PRINT"Was a card shown?  -  Press Y/N"
12 R$=INKEY$:IFR$="Y"THEN20ELSEIFR$="N"THEN17ELSEIFR$="E"THEN88ELSE12
13 IFPEEK(Z+A*21+P)=AORPEEK(Z+A*21+I)=AORPEEK(Z+A*21+S)=ATHEN15ELSEPRINTA$(C1):P
RINT:PRINT"Say  -  Sorry  "A$(U)", I can't help you.":PRINT:PRINT"Press space ba
r to continue"
14 R$=INKEY$:IFR$=""THEN14ELSEIFR$="E"THEN88ELSE18
15 PRINT:PRINT:PRINTA$(C1)", Please show your card to "A$(U):PRINT:PRINT:PRINT"P
ress space bar to continue"
16 R$=INKEY$:IFR$=""THEN16ELSEIFR$="E"THEN88ELSE21
17 POKEZ+A*21+P,7:POKEZ+A*21+I,7:POKEZ+A*21+S,7
18 W=W+1:IFW=N-1THEN21ELSET=T+1:IFT>NTHENT=1
19 GOTO10
20 IFU<>C1THENGOSUB56ELSEPRINT:INPUT"Enter card shown";O:IFO<>PANDO<>IANDO<>STHE
N20ELSEIFPEEK(Z+A*21+O)=ATHEN21ELSEPOKEZ+10,INT((Z+A*21)/256):POKEZ+9,Z+A*21-INT
((Z+A*21)/256)*256:POKEZ+11,A:POKE16526,189:POKE16527,115:X=USR(Z+A*21+O)
21 CLS:GOSUB30:GOSUB31:GOSUB23:GOSUB30:GOSUB31:GOSUB26:GOSUB30:GOSUB31
22 GOTO73
23 POKE16526,25:POKE16527,116:POKEZ+159,6:IFA1=0THENPOKEZ+157,6:X=USR(Z+22)
24 IFA2=0THENPOKEZ+157,6:X=USR(Z+28)
25 IFA3<>0THENRETURNELSEPOKEZ+157,9:POKEZ+159,9:X=USR(Z+34):RETURN
26 POKE16526,108:POKE16527,116:IFA1>0THENPOKEZ+157,6:X=USR(Z+22)
27 IFA2>0THENPOKEZ+157,6:X=USR(Z+28)
28 IFA3>0THENPOKEZ+157,9:X=USR(Z+34)
29 RETURN
30 POKE16526,197:POKE16527,115:X=USR(Z+22):RETURN
31 IFA1>0THEN32ELSEPOKE16526,115:POKE16527,114:X=USR(6):A1=X
32 IFA2>0THEN33ELSEPOKE16526,168:POKE16527,114:X=USR(6):A2=X
33 IFA3>0THENRETURNELSEPOKE16526,179:POKE16527,114:X=USR(9):A3=X:RETURN
34 ONC2(C3)GOTO35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55
35 C4=15488:RETURN
36 C4=15552:RETURN
37 C4=15616:RETURN
38 C4=15680:RETURN
39 C4=15744:RETURN
40 C4=15808:RETURN
41 C4=15511:RETURN
42 C4=15575:RETURN
43 C4=15639:RETURN
44 C4=15702:RETURN
45 C4=15766:RETURN
46 C4=15830:RETURN
47 C4=15531:RETURN
48 C4=15595:RETURN
49 C4=15659:RETURN
50 C4=15723:RETURN
51 C4=15787:RETURN
52 C4=15851:RETURN
53 C4=15915:RETURN
54 C4=15979:RETURN
55 C4=16043:RETURN
56 IFPEEK(Z+A*21+P)=AORPEEK(Z+A*21+I)=AORPEEK(Z+A*21+S)=ATHENRETURNELSEPOKEZ+1,P
:POKEZ+3,I-P:POKEZ+13,S-I:POKEZ+5,0:POKEZ+11,A:POKE16526,202:POKE16527,114:X=USR
(Z+A*21):RETURN
57 PRINT@772,CHR$(31);:INPUT"Number of Players";N$:N=VAL(N$):IFN<2ORN>6THENPRINT
@772,"2 to 6  Please."CHR$(30):FORX=1TO1500:NEXT:GOTO57ELSEPOKEZ+20,N:POKEZ+17,2
1*N+1:RETURN
58 PRINT@896,CHR$(31);:INPUTN$:IFVAL(N$)>0ANDN$<>"0"THEN58ELSEC2(C3)=VAL(N$)
59 IFC2(C3)>21THEN58ELSEIFC2(C3)<>0THENGOSUB34:POKEC4,94:C3=C3+1:PRINT@896,CHR$(
30):GOTO58
60 CLS:PRINT@330,"Are these cards correct?  -  Press Y/N":PRINT:FORX=1TOC3-1:PRI
NTTAB(10)N$(C2(X)):NEXT
61 R$=INKEY$:IFR$="N"THENV=0:GOTO6ELSEIFR$<>"Y"THEN61
62 FORX=1TOC3-1:FORO=1TON:POKEZ+O*21+C2(X),7:NEXT:POKEZ+C1*21+C2(X),C1:NEXT:GOTO
87
63 S3=INT(18/N):FORO=1TON:POKE0+O,S3:NEXT:S3=S3+1:IFN=4THEN64ELSEIFN=5THEN66ELSE
RETURN
64 ONO2GOTO65,65,66,67
65 POKEQ+D2+1,S3:POKEQ+D2+2,S3:RETURN
66 POKEQ+4,S3:POKEQ+1,S3:RETURN
67 POKEQ+1,S3:POKEQ+2,S3:RETURN
68 ONO2GOTO69,69,70,71,72
69 POKEQ+D2+1,S3:POKEQ+D2+2,S3:POKEQ+D2+3,S3:RETURN
70 POKEQ+4,S3:POKEQ+5,S3:POKEQ+1,S3:RETURN
71 POKEQ+5,S3:POKEQ+1,S3:POKEQ+2,S3:RETURN
72 POKEQ+1,S3:POKEQ+2,S3:POKEQ+3,S3:RETURN
73 IFA1>0ANDL1=0THENP$=MID$(N$(A1),3):X$="I've found the murderer!!":X1$="The mu
rderer - ":L1=1
```

```
74 IFA2>0ANDL2=0THENI$=MID$(N$(A2),4):Y$="I've found the murder weaPon!!":Y1$="T
he murder weaPon -  ":L2=1
75 IFA3>0ANDL3=0THENS$=MID$(N$(A3),4):Z$="I've found the murder room!!":Z1$="The
 murder room -  ":L3=1
76 IFL1+L2+L3=3THENE$="To 9uit the Pro9ram Press  Q"+CHR$(226)+" To return to d
isPlay Press  D"
77 CLS:PRINT@128,X$:PRINT:PRINTY$:PRINT:PRINTZ$:PRINT@724,"For disPlay  -  Press
 D":PRINT@788,"For next Player  -  Press N"
78 R$=INKEY$:IFR$="D"THEN79ELSEIFR$="N"THEN87ELSEIFR$="E"THEN88ELSE78
79 CLS:PRINTTAB(12)"CM  PP  RG  MP  MS  MW      DA  CA  RE  RO  LP  SP":M=11:FOR
O=1TON:D$=LEFT$(A$(O),9):PRINTD$;:FORX=1TO6:PRINTTAB(M)PEEK(Z+O*21+X);:M=M+4:NEX
TX:M=M+4:FORX=7TO12:PRINTTAB(M)PEEK(Z+O*21+X);:M=M+4:NEXTX:M=11:PRINT:NEXTO:PRIN
T
80 PRINTTAB(12)"HA  LO  DR  KI  BA  CO  BI  LI  ST":M=11:FORO=1TON:D$=LEFT$(A$(O
),9):PRINTD$;:FORX=13TO21:PRINTTAB(M)PEEK(Z+O*21+X);:M=M+4:NEXTX:M=11:PRINT:NEXT
O:IFL1+L2+L3>0THENPRINT@963,"For answers  -  Press A";
81 PRINT@995,"Press sPace bar to continue";
82 R$=INKEY$:IFR$=""THEN82ELSEIFR$="A"THEN83ELSEIFR$="E"THEN88ELSEIFR$="Q"THEN85
ELSE87
83 CLS:PRINT@460,X1$;P$:PRINT@524,Y1$;I$:PRINT@588,Z1$;S$:PRINT@866,E$:PRINT@994
,"Press sPace bar to continue";
84 R$=INKEY$:IFR$=""THEN84ELSEIFR$="Q"THEN85ELSEIFR$="D"THEN79ELSEIFR$="E"THEN88
ELSE87
85 CLS:PRINT@460,"Do you want another 9ame?"
86 R$=INKEY$:IFR$=""THEN86ELSEIFR$="Y"THENRUN2ELSEEND
87 POKE16526,191:POKE16527,116:X=USR(0):GOTO6
88 POKE16526,203:POKE16527,116:X=USR(0):A1=0:A2=0:A3=0:L1=0:L2=0:L3=0:X$="":Y$="
":Z$="":X1$="":Y1$="":Z1$="":P$="":I$="":S$="":E$="":T$="**  RePlay last hand  *
*":GOTO6
89 CLS:PRINTCHR$(23):PRINT@778,"Loadin9  Data ....":FORX=1TO21:READN$:NEXT:FORX=
29299TO29907:READA:POKEX,A:T=T+A:IFT>255THENT=T-256
90 NEXT:IFT<>156THENPRINT@842,"Incorrect Data":ENDELSERESTORE:RETURN
91 CLS:PRINTCHR$(23):FORX=0TO125:SET(X,1):NEXT:FORO=1TO31:SET(125,O):NEXT:FORX=1
24TO0STEP-1:SET(X,31):NEXT:FORO=31TO1STEP-1:SET(0,O):NEXT:PRINT@344,"CLUEDO":RE
TURN
92 FORX=1TO21:READN$(X):NEXT:POKE16526,177:POKE16527,115:X=USR(Z):X=USR(Z+155):P
OKEZ+7,8:PRINT@770,"Do you want instructions?"
93 R$=INKEY$:IFR$="N"THENRETURNELSEIFR$<>"Y"THEN93
94 CLS:PRINTTAB(26)"CLUEDO":PRINT:PRINT"Introduction":PRINT"This Pro9ram is not
 a 9ame in itself but is used within one. It is desi9ned to be used with the PoPu
lar board 9ame of the same  name, Produced by Murfett."
95 PRINT"The comPuter will Play one Player's hand in the 9ame and will  come uP
 with (hoPefully) the riaht answers. (as well as catchin9the cheats!)":PRINT:PRI
NT"ExPlanation of inPuts:
96 PRINT"Number of Players: from 2 to 6":PRINT"ComPuter Player's number: any num
ber uP to the total Players.":PRINT"( Players are numbered in a clock-wise direc
tion.)":PRINT"Who's to deal: what is dealer's number?"
97 IFINKEY$=""THEN97ELSECLS:PRINT"Durin9 the 9ame:":PRINT:PRINT"Firstly, you mus
t understand the board 9ame itself.":PRINT:PRINT"The Player usin9 the comPuter w
ill need to familiarise himself  with the DISPLAY which is available after each
 Player's turn."
98 PRINT"As the 9ame develoPs the Player will obtain clues as to where tomove on
 the board.":PRINT"The Pro9ram will ask for information to be 9iven on what is h
aPPenin9 - i.e. who is callin9, was a card shown, etc."
99 PRINT"If incorrect information is entered durin9 the Play, Press E at any tim
e and the hand will be re-Played.":PRINT"To 9uit the Pro9ram Press Q"
100 IFINKEY$=""THEN100ELSECLS:PRINT"The DISPLAY -   CM   PP   RG   MP   MS   MW"
:PRINTTAB(10)"Tom   7    7    1    0    1    7":PRINTTAB(10)"Mary  2    7    7
  0    7    7"
101 PRINTTAB(10)"Harry 7    0    7    7    7    7":PRINTTAB(10)"Sue   7    7
7    9    7    4":PRINT
102 PRINT"The samPle disPlay above shows information about the PERSON":PRINT"9ro
uP of cards.(CM is Col Mustard, PP is Prof Plum etc)":PRINT"The disPlay shows th
at Tom (Player 1) has Rev Green and Miss S. Mary (2) has CM and Sue (♦) has MW."
103 PRINT"A number 7 means that the Player does not hold that card  i.e.  neithe
r Tom, Mary or Sue hold Prof Plum as they all have a 7 forthat card.":PRINT"Thus
 it must be Harry's or it's the murderer."
104 IFINKEY$=""THEN104ELSEPRINT@384,CHR$(31);:PRINT"A zero means - I don't know
 yet.":PRINT"The 9 for Sue means that she holds MP or a card in another 9rouP- (a
lso shown with a 9). These numbers will be chan9ed as more  deductions are made.
":PRINT
105 PRINT"After several Practice runs the disPlay will be very useful.":PRINT"Wh
en answers are found this will be indicated after each hand.":PRINT:PRINT"Good l
uck!";
106 IFINKEY$=""THEN106ELSE91
107 CLS:FORX=1TON:PRINT"Enter Player "X"'s name ";:INPUTA$(X):PRINT:NEXT:GOTO6
108 DATA1. Col Mustard,2. Prof Plum,3. Rev Green,4. Mrs Peacock,5. Miss Scarlett
,6. Mrs White,7. Da99er,8. Candlestick,9. Revolver,10. RoPe,11. Lead PiPin9,12.
SPanner
109 DATA13. Hall,14. Loun9e,15. Dinin9 Room,16. Kitchen,17. Ball Room,18. Conser
vatory,19. Billiard Room,20. Library,21. Study
110 DATA205,127,10,36,221,33,49,117,17,21,0,237,75,68,117,65,221,229,253,225,221
,126,21,254,7,32,13,13,40,4,221,25,24,242,108,38,0,195,154,10,45,40,248,72,253,3
5,36,253,229,221,225,24,223,205,127,10,221,33,55,117,38,7,24,200,205,127,10
111 DATA221,33,61,117,38,13,24,189,42,57,117,237,91,49,117,25,126,254,7,201,205,
127,10,34,57,117,237,91,49,117,74,205,197,114,32,6,12,121,254,2,40,27,58,53,117,
60,50,53,117,254,3,40,96,254,2,40,6,237,91
112 DATA51,117,24,224,237,91,61,117,24,218,205,190,114,32,14,237,91,51,117,205,1
97,114,32,5,237,91,61,117,25,34,63,117,126,186,40,117,70,90,42,57,117,28,123,254
,22,40,106,35,120,190,32,245,114,24,242,237,91,59,117,33,198,117,25,52,229,193,3
3
113 DATA208,117,25,10,190,192,42,57,117,20,122,254,22,200,35,126,187,40,246,54,7
,24,242,205,190,114,40,2,186,192,237,91,51,117,205,197,114,40,2,186,192,237,91,6
1,117,205,197,114,40,2,186,192,42,55,117,35,34,55,117,77,205,190,114,40,1,113
114 DATA237,91,51,117,205,197,114,40,1,113,237,91,61,117,205,197,114,200,113,201
,42,63,117,1,235,255,58,59,117,61,40,3,9,24,250,58,68,117,1,21,0,54,7,61,40,3,9,
24,248,42,63,117,58,59,117,119,195,40,115,205,127,10,14,155,54,0,35,13,32,250
115 DATA201,205,127,10,22,0,195,14,115,205,127,10,229,221,225,58,55,117,237,75,6
5,117,254,8,200,237,177,40,6,221,229,225,61,24,239,43,34,63,117,35,237,177,40,24
1,8,42,63,117,229,253,225,17,187,138,25,30,235,125,36,254,21,40,6,250,1,116,131
116 DATA24,245,79,253,43,61,32,251,253,34,57,117,124,50,59,117,33,69,117,9,205,1
53,115,8,24,192,205,127,10,62,0,50,215,117,205,97,116,126,186,40,29,254,7,250,63
,116,32,22,9,29,32,241,33,205,117,53,40,18,253
117 DATA229,225,35,24,226,33,215,117,52,24,238,253,229,221,225,24,232,58,207,117
,61,33,215,117,190,192,58,68,117,221,229,225,54,7,61,200,9,24,249,1,21,0,237,91,
68,117,229,253,225,201,205,127,10,205,97,116,126,254,7,40,22,9,29,32,247,58
118 DATA68,117,61,186,40,14,33,205,117,53,200,253,229,225,35,24,226,20,24,231,23
7,91,68,117,253,229,225,20,126,254,7,32,3,9,24,247,186,40,223,24,63,117,122,50,5
9,117,22,0,33,48,117,9,61,32,252,24,57,117,42,63,117,205,14,115,195,130,116
119 DATA33,70,117,17,218,117,1,185,0,237,176,201,33,218,117,17,70,117,195,197,11
6
```

STOCKMARKET

Stockmarket is a game I wrote a few months ago.

Lines	Function
10-300	Introduction
310-450	Displays table
550-660	Buying shares
670-720	Buying more shares
750-1180	Select a change in trends
1190-1210	Print it
1220-1370	Sell shares
1380-1470	Assign values for graph
1480-1670	Graph routine
1680-1730	Find status (up/down)
1740-1900	Quit routine

This simulation is designed for the System 80/TRS-80 machines and uses PRINT@ and PRINTUSING otherwise I see no problems in conversion to other machines. The program runs on a 64 character screen and occupies about 9K.

David Thomas
Yandina Qld

```
10 REM STOCK MARKET
20 CLS:CLEAR1000:DEFINTA-Z:WE=0:F$(1)="-":F$(2)="+":F$(3)=CHR$(34):DIMB$(8)
30 FORX=1TO8:VV(X)=0:NEXTX:L$="$$######.##"
40 PRINT@0,STRING$(64,"*");
50 FORC=64TO448STEP64:PRINT@C,"*";:PRINT@C+63,"*";:NEXTC
60 FORC=449TO511:PRINT@C,"*";:NEXTC
70 PRINT@88,"* STOCK MARKET *";
80 PRINT@152,"****************";
90 PRINT@216,"BY  DAVID THOMAS";
100 PRINT@323,"Enter '0' if you accidently Press buy or sell and you want";
110 PRINT@387,"neither.";
120 FORC=1TO700:NEXTC
130 PRINT@532,"";
140 INPUT"How many Players (1 to 5)";A
150 IFA(1ORA)15PRINTCHR$(27)CHR$(29);:GOTO130
160 DIMA$(A)
170 FORB=1TOA
180 PRINT@660,"";
190 PRINT"Player ";B;"'s name is";:INPUTA$(B)
200 PRINTCHR$(27)CHR$(29);
210 NEXTB
220 PRINT@707,"";:INPUT"Delay between 9raPhin9 (e9. 10 = 1 9raPh every 10 9oes)"
;TR
230 IFTR<1ORTR>40PRINT@771,"Try a9ain , Please";:GOTO220
240 CLS:FORB=1TOA:PRINT"Player#";B;" is "A$(B):NEXTB
250 PRINT"Press <newline> to continue"
260 I$=INKEY$:IFI$=CHR$(13)GOTO270ELSEGOTO260
270 FORC=1TO8:READB$:C$(C)=B$:NEXTC
280 FORC=1TOA:M(C)=1000:NEXTC
290 FORC=1TO8:READD:D(C)=D:NEXTC .
300 K=1
310 CLS:PRINT@0," COMPANY","PRICE","STATUS"," ASSETS"
320 FORP=1TO8:PRINT@P%64,P;C$(P),D(P),,VH(P);"(";VV(P);")";:NEXTP
330 FORC=98TO588STEP64
340 GOSUB1680
350 PRINT@C,F$(IU);
360 NEXTC
370 PRINT:PRINTA$(K)"'s turn":PRINT"Your liquid assets are:";:PRINTUSINGL$;M(K)
380 PRINT"'B' BUY ";STRING$(9,32);"'S' SELL";STRING$(8,32);"'N' NO TRADING";STRI
NG$(9,32);"'Q' QUIT";
390 PRINT"WELL?";
400 I$=INKEY$:IFI$=""GOTO400
410 IFI$<>"B"ANDI$<>"S"ANDI$<>"N"ANDI$<>"Q"GOTO400
420 IFI$="Q"GOTO1740
430 IFI$="B"GOTO460
440 IFI$="N"GOTO470
450 IFI$="S"GOTO530
460 GOSUB550
470 GOSUB750
480 K=K+1:IFK=A+1K=1
490 WE=WE+1:IFWE=TRGOTO500:ELSEGOTO520
500 GOSUB1480
510 WE=0
520 GOTO310
530 GOSUB1220
540 GOTO480
550 PRINT:INPUT"WHICH COMPANY (1 TO 8)";NM
560 IFNM=0RETURN
570 IFNM(1ORNM>8PRINTCHR$(27)CHR$(27)CHR$(29);:GOTO550
580 IFK=VV(NM)THEN670
590 IFVV(NM)<>0THENPRINT"ALREADY BOUGHT":FORX=1TO300:NEXTX:RETURN:ELSE600
600 INPUT"HOW MANY SHARES";SH
610 IFSH=0RETURN
620 IFM(K)-(SH*D(NM))<1PRINT"YOU HAVE'NT ENOUGH MONEY";:PRINTCHR$(27)CHR$(29);:G
OTO600
630 VV(NM)=K
640 M(K)=M(K)-(SH*D(NM))
650 VH(NM)=SH
660 RETURN
670 INPUT"HOW MANY MORE SHARES";SH
680 IFSH=0RETURN
690 IFM(K)-VH(NM)-(SH*VV(NM))<0.01PRINT"YOU HAVE'NT ENOUGH MONEY";:PRINTCHR$(27)
CHR$(29);:GOTO670
700 VH(NM)=VH(NM)+SH
710 M(K)=M(K)-(SH*D(NM))
720 RETURN
```

```
730 DATA KMART,ALCOA,APPLE,AMPOL,MT ISA MINES,BHP,AMOCO,TAA
740 DATA 40,30,60,70,60,80,50,90
750 R=RND(10):IFR>2ANDR<8GOTO990
760 R=RND(15):ONRGOTO770,790,970,810,830,970,850,870,970,890,910,970,930,950,970
770 O=RND(19)+RND(10):G$="APPLE WISHES TO ADVISE ALL STOCKHOLDERS.....  OF A "
+STR$(O)+"% RISE   IN PROFITS FOR 1982":D(3)=D(3)+(O/100*D(3)):XZ(3)=2
780 GOTO1190
790 O=RND(20)+7:G$="KMART ANNOUNCES.....   A "+STR$(O)+"% DEPRECIATION IN S
HARE PRICES.....   DUE TO INCREASED OPERATING COSTS.....":D(1)=D(1)-(O
/100*D(1)):XZ(1)=1
800 GOTO1190
810 O=RND(30):G$="BHP'S DEFICIT FOR 1982 IS $"+STR$(O)+" 000 000.....  DU
E TO STRIKES THROUGHOUT 1982.....":D(6)=D(6)-(O/100*D(6)):XZ(6)=1
820 GOTO1190
830 O=RND(5):G$="NO."+STR$(O)+" SHAFT..... AT MT. ISA..... HAS CLOSED DOWN BEC
AUSE OF A CAVE-IN..... MARKET FUTURE UNCERTAIN.....":D(5)=D(5)-((O*10)
/100*D(5)):XZ(5)=1
840 GOTO1190
850 G$="NO REPORTS.....  MARKET STABLE....."
860 GOTO1190
870 O=RND(5):PO=RND(20):G$="PETROL RISES BY "+STR$(O)+" CENTS.....   A "+STR
$(PO)+"%  BONUS RESULTS TO AMPOL SHAREHOLDERS.....":D(4)=D(4)+((PO/100)*D(4)):
XZ(4)=2
880 GOTO1190
890 G$="NO REPORTS..... MARKET STABLE....."
900 GOTO1190
910 O=RND(30)+10:G$="ALCOA IS PLEASED TO ANNOUNCE.....     A BOOM IN THE AL
UMINIUM INDUSTRY.....  SHARES RISE BY "+STR$(O)+"% !!!!.....":D(2)=D(2)+((O/1
00)*D(2)):XZ(2)=2
920 GOTO1190
930 G$="AIRFAIRS SOAR !!!.....  'TAA'.....":D(8)=D(8)-((RND(10)+5)/
100*D(8)):XZ(8)=1
940 GOTO1190
950 O=RND(20)+10:G$="AMOCO..... NEW WELL SUNK..... "+STR$(O)+"% INCREASE IN
 PRODUCTION":D(7)=D(7)+((O/100)*D(7)):XZ(7)=2
960 GOTO1190
970 G$="NO REPORTS.....     MARKET STABLE....."
980 GOTO1190
990 R=RND(15):ONRGOTO1010,1030,1180,1050,1080,1180,1100,1180,1120,1140,1180,1180
,1160,1180
1000 GOTO1190
1010 O=RND(20):G$="TAA ANNOUNCES.....    A CLEAR PROFIT OF $"+STR$(O)+" 0
00 000.....  FOR THIS FINANCIAL YEAR.....":D(8)=D(8)+((O/100)*D(8)):XZ(8)
=2
1020 GOTO1190
1030 O=RND(10)+RND(5):G$="AMOCO ANNOUNCES.....    A "+STR$(O)+"% DROP IN SHARE
VALUE.....":D(7)=D(7)-((O/100)*D(7)):XZ(7)=1
1040 GOTO1190
1050 O=RND(40)-RND(10):G$="BHP..... EUREKA!!!.....  A NEW VEIN HAS BEE
N STRUCK..... A 2 FOR 1 BONUS IS AVAILABLE FOR EXISTING SHAREHOLDERS.....
...":IFVV(6)=KTHEND(6)=D(6)*2:ELSED(6)=D(6)+((O/100)*D(6)):XZ(6)=2
1060 XZ(6)=2
1070 GOTO1190
1080 O=RND(20)+RND(10)+5:G$="MT ISA..... LEAD PRICES RISE!!!..... DUE
 TO SHORTAGE.....":D(5)=D(5)+((O/100)*D(5)):XZ(5)=2
1090 GOTO1190
1100 O=RND(10):G$="AMPOL..... TANKERS REFUSE TO LOAD OR UNLOAD AMPOL FUEL.....
...  PAY RISE CLAIM.....":D(4)=D(4)-((O/100)*D(4)):XZ(4)=1
1110 GOTO1190
1120 O=RND(10)+RND(5):G$="APPLE..... NEW TECHNOLOGY.....256K chip ALREADY ON T
HE MARKET.....SLUMP IN MICRO SALES.....":D(3)=D(3)-((O/100)*D(3)):XZ(3)=1
1130 GOTO1190
1140 O=RND(10)+RND(5)+RND(0):G$="BAUXITE GLUT..... ALCOA REDUCES PRICES B
Y "+STR$(O)+"% ..... REFINERY'S REFUSE ANY MORE SHIPMENTS.....":D(2)=D(2)-((O/10
0)*D(2)):XZ(2)=1
1150 GOTO1190
1160 O=RND(20)+10:G$="KMART SHARES INCREASE IN VALUE.....  BY "+STR$(O)+"%!!
!.....   BECAUSE OF BULK BUYING OF ITEMS.....":D(1)=D(1)+((O/100)*D(1
)):XZ(1)=2
1170 GOTO1190
1180 G$="NO REPORTS..... MARKET STABLE....."
1190 PRINT@768,CHR$(31);:FORLL=1TOLEN(G$):PRINT@768+LL,MID$(G$,LL,1);:FOROP=1TO2
0:NEXTOP:NEXTLL
1200 G$=""
1210 FOROP=1TO500:NEXTOP:RETURN

1220 PRINT:INPUT"WHICH COMPANY <1 TO 8)";WW
1230 IFWW=0RETURN
1240 IFWW(1ORWW)8PRINTCHR$(27)CHR$(27)CHR$(29);:GOTO1220
1250 IFVV(WW)<KPRINT"YOU DON'T OWN ";C$(WW);:PRINTCHR$(27)CHR$(27)CHR$(29);:GOT
O1220
1260 IFWW=0RETURN
1270 IFWW(1ORWW)8PRINTCHR$(27)CHR$(29);:GOTO1220
1280 INPUT"HOW MANY SHARES <0 = ALL)";SH
1290 IFSH<>0GOTO1330
1300 IFSH=0THENVV(WW)=0:M(K)=M(K)+(D(WW)*VH(WW)):VH(WW)=0
1310 IFVH(WW)VAL0THENVV(WW)=0
1320 RETURN
1330 IFSH>VH(WW)PRINT"YOU DON'T OWN THAT MANY SHARES";:PRINTCHR$(27)CHR$(29);:GO
TO1280
1340 M(K)=M(K)+(D(WW)*SH)
1350 VH(WW)=VH(WW)-SH
1360 IFVH(WW)=0THENVV(WW)=0
1370 RETURN
1380 U(1)=D(1):U(2)=D(2):U(3)=D(3):U(4)=D(4):U(5)=D(5):U(6)=D(6):U(7)=D(7):U(8)=
D(8)
1390 M=U(1):IFU(1)>MTHENM=U(1)
1400 IFU(2)>MTHENM=U(2)
1410 IFU(3)>MTHENM=U(3)
1420 IFU(4)>MTHENM=U(4)
1430 IFU(5)>MTHENM=U(5)
1440 IFU(6)>MTHENM=U(6)
1450 IFU(7)>MTHENM=U(7)
1460 IFU(8)>MTHENM=U(8)
1470 RETURN
1480 REM
1490 GOSUB1380
1500 CLS
1510 N=8
1520 L=M
1530 FORI=191TO767STEP64
1540 PRINT@I,L;TAB(11)"-";CHR$(149)
1550 L=L-M/10:NEXTI
1560 PRINT@832,0;TAB(12)CHR$(141);
1570 PRINT@845,STRING$(50,140)
1580 PRINT@128,"          KMART";CHR$(143);"ALCOA ";CHR$(143);"APPLE";CHR$(143
);"AMPOL";CHR$(143);"M.I.M";CHR$(143);"B H P";CHR$(143);"AMOCO";CHR$(143);"T A A
";
1590 FORC=0TO7
1600 X=D(C+1)
1610 FORY=0TOINT(X/M*30+.5)
1620 FORZ=0TOINT(100/N)-2
1630 SET(INT(100/N)*C+Z+26,40-Y)
1640 NEXTZ:NEXTY:NEXTC
1650 PRINT@960,"PRESS <NEWLINE) TO CONTINUE"
1660 I$=INKEY$:IFI$=CHR$(13)GOTO1670:ELSE1660
1670 CLS:RETURN
1680 ' STATUS FINDER
1690 IFC=98Q=1ELSEIFC=162Q=2ELSEIFC=226Q=3ELSEIFC=290Q=4ELSEIFC=354Q=5ELSEIFC=41
8Q=6ELSEIFC=482Q=7ELSEIFC=546Q=8
1700 IFXZ(Q)=0THENIU=3:RETURN
1710 IFXZ(Q)=1THENIU=1:RETURN
1720 IFXZ(Q)=2THENIU=2:RETURN
1730 RETURN
1740 CLS
1750 FORCO=1TOR
1760 YY(CO)=MO(CO)
1770 FORNO=1TO8
1780 IFVV(NO)=COTHENYY(CO)=YY(CO)+D(NO)*VH(NO)
1790 NEXTNO,CO
1800 PRINT@0,STRING$(64,140);
1810 FORQQ=1TOR
1820 PRINT@128,"PLAYER      LIQUID ASSETS   INVESTMENTS          TOTAL";
1830 PRINT@192,"------      -------------   -----------          -----";
1840 PRINT@192+(64*QQ),A$(QQ),M(QQ),YY(QQ),YY(QQ)+M(QQ);
1850 NEXTQQ
1860 PRINT@192+(64*QQ),STRING$(64,140);
1870 PRINT:PRINT:INPUT"ANOTHER GAME ";OI9$
1880 IFLEFT$(OI9$,1)="Y"THEN10
1890 CLS:PRINT"BYE FROM THE STOCKMARKET !"
1900 END
```

CADDYSHACK

Caddyshack is a simple golf-playing routine. The player swings or putts by entering numbers, which then become the stroke lengths. The ball lands somewhere else on the green; if you've estimated your swing well enough it will land in the cup and you stroll on to the next hole. A running score is kept up throughout the play, and presented regularly.

Much can be done to improve the program. For a start, it's not noisy enough. The 'goodbye' routine (lines 470 to 500 inclusive) uses sound, certainly, but the WIN routine could do with a bit of racket. Try this: 380 FOR WIN=0 TO 9: GOSUB 520 520 FOR NOISE=600 TO 750 STEP 5: SOUND NOISE,2: NEXT NOISE: RETURN

You might like the machine to sound a beep at the beginning of each hole – just add this to line 20: 20 FOR HOLE=1 TO 9: BEEP

You might like to change the PAUSE values.

Use is made throughout the program of the delay subroutine at line 510. Note how the delay value, PAUSE, is set at the subroutine calls, so that the length varies at each call. Thus, the delay initiated at line 400 will run fifty times, while the pause at line 490 will be two hundred runs long.

Neville Predebon
West Preston VIC

```
5 REM - Caddyshack -
7 CLS
10 C=0
20 FOR HOLE=1 TO 9
30 J=INT(RND(0)*12)-1
40 SC=0
50 PAR=INT(RND(0)*4)+2
60 GOSUB 270
70 INPUT "Stroke";STROKE: CLS
80 IF J>24 THEN STROKE=-STROKE
90 J=J+INT(STROKE/INT(RND(0)*PAR))
100 GOSUB 270
110 SC=SC+1
120 PRINT "After that stroke, your score is now";SC
130 IF J<>24 THEN 70
140 GOSUB 350
150 C=C+SC
160 CLS: PRINT: PRINT TAB(4);"Your score for";HOLE;"hole/s is";SC
170 PAUSE=999: GOSUB 510
180 FOR LIN=1 TO 29
190 FOR SPACE=1 TO LIN
200 PRINT " ";
210 NEXT SPACE
220 PAUSE=39: GOSUB 510
230 PRINT "Stand by-"
240 NEXT LIN
250 NEXT HOLE
260 GOTO 420
270 IF J>30 THEN J=30
280 FOR NOISE=415 TO 545 STEP 5: SOUND NOISE,2: NEXT NOISE
290 PRINT TAB(3);"Par for this hole >-";PAR
300 PRINT: PRINT TAB(3);"### HOLE NUMBER";HOLE;"###": PRINT: PRINT
310 FOR PUTT=1 TO J-1
320 PRINT " ";
330 NEXT PUTT
340 PRINT "o"
350 PRINT "------------------------ ----------------"
360 RETURN
370 PRINT "------------------------o----------------"
380 FOR WIN=0 TO 9
390 PRINT: PRINT TAB(7);"You got it in";SC;"strokes!"
400 PAUSE=49: GOSUB 510
410 NEXT WIN: RETURN
420 PRINT TAB(4);"That's the end of that round."
430 PRINT: PRINT TAB(4);"You scored";C
440 PRINT TAB(4);"Your average was";INT(C/9)
450 PRINT: INPUT "Would you like to play another round";ANS$
460 IF LEFT$(ANS$,1)="y" OR LEFT$(ANS$,1)="Y" THEN RUN
470 CLS: PRINT @ 174,"GOODBYE!"
480 FOR NOISE=830 TO 1046 STEP 4: SOUND NOISE,3: NEXT NOISE
490 PAUSE=199: GOSUB 510: CLS
500 FOR NOISE=1046 TO 830 STEP-4: SOUND NOISE,3: NEXT NOISE: GOTO 470
510 FOR DELAY=0 TO PAUSE: NEXT DELAY: RETURN
```

ALIEN WIPEOUT

```
10 REM************************************************************
20 REM**                A L I E N  W I P E O U T                **
30 REM**                  By Wayne McCullough                    **
40 REM**                       Darwin N.T.                       **
50 REM**            Originally written 16th May 1982            **
60 REM**                  Revised 8th October 1983              **
70 REM************************************************************
80 CLEAR1000:HS=0
90 CLS:PRINTTAB(20);"A l i e n  W i p e o u t":PRINTTAB(20);STRING$(25,134):GOSU
B990
100 A1$=CHR$(153)+CHR$(191)+CHR$(157)+CHR$(145):A2$=CHR$(166)+CHR$(187)+CHR$(179
)+CHR$(132)
110 A3$=CHR$(156)+CHR$(183)+CHR$(157)+CHR$(148):A4$=CHR$(140)+CHR$(183)+CHR$(157
)+CHR$(132)
120 A5$=CHR$(156)+CHR$(175)+CHR$(141)+CHR$(148):A6$=CHR$(152)+CHR$(183)+CHR$(157
)+CHR$(144)
130 A7$=CHR$(134)+CHR$(189)+CHR$(151)+CHR$(132):A8$=CHR$(157)+CHR$(183)+CHR$(187
)+CHR$(174)
140 A9$=CHR$(166)+CHR$(157)+CHR$(183)+CHR$(132):B1$=CHR$(182)+CHR$(190)+CHR$(182
)+CHR$(148)
150 B2$=CHR$(158)+CHR$(191)+CHR$(173):B3$=CHR$(140)+CHR$(179)+CHR$(166)+CHR$(153
)+CHR$(132)
160 B4$=CHR$(141)+STRING$(2,153)+CHR$(133):B5$=CHR$(156)+CHR$(179)+CHR$(153)+CHR
$(133)
170 B6$=CHR$(167)+CHR$(183)+CHR$(133):B7$=CHR$(174)+CHR$(179)+CHR$(157):B8$=CHR$
(153)+CHR$(166):B9$=CHR$(155)+CHR$(167)
180 C1$=CHR$(182)+CHR$(180)+CHR$(182)+CHR$(148):C2$=CHR$(166)+CHR$(167)+CHR$(132
):C3$=STRING$(2,166)+CHR$(132)
190 C4$=CHR$(140)+CHR$(174)+CHR$(179)+CHR$(191)+CHR$(179)+CHR$(191)+CHR$(179)+CH
R$(157)+CHR$(140)
200 PRINTA1$;" ";A2$;" ";A3$;" ";A4$;" ";A5$;" ";A6$;" ";A7$;" ";A8$;" ";A9$;" "
;B1$;" ";B2$;" ";B3$;" ";B4$
210 PRINT@256,"Aliens have invaded the Earths skys, you must aim your photon  l
aser and destroy as many aliens as possible in the sixty"
220 PRINT"seconds which the Earth has left."
230 PRINT"Use the arrow keys to aim, and the space bar to fire. Good luck!";
240 FORSOS=1TO4:GOSUB1050:NEXTSOS:POKE16396,175:REM DISABLE BREAK
250 PRINT@571,CHR$(176);CHR$(140);CHR$(131);CHR$(131);:PRINT@633,CHR$(160);CHR$(
134);CHR$(152);CHR$(131);CHR$(134);CHR$(164);
260 PRINT@697,CHR$(138);CHR$(146);CHR$(140);CHR$(131);CHR$(140);CHR$(129);:PRINT
@762,CHR$(130)+CHR$(137)+CHR$(164)+CHR$(176)+CHR$(176)
270 PRINT@590,C4$;:FORX=48TO116:SET(X,28):RESET(X-1,28):NEXTX
280 PRINT@832,"SIR! We have recieved a communication from the alien mothership":
PRINTCHR$(34);"Prepare to die human!";CHR$(34)
290 GOSUB1090
300 FA$="/      "+CHR$(92)+CHR$(26)+STRING$(7,24)+CHR$(92)+"      /":FB$=CHR$(92)+
"      /"+CHR$(26)+STRING$(7,24)+"/      "+CHR$(92):FC$=STRING$(7,32)+CHR$(26)+STR
ING$(7,24)+STRING$(7,32)
310 E5$="":E3$="":E4$="":FORX1=1TO7:E3$=E3$+CHR$(RND(30)+161)+" ":NEXTX1:FORX1=1
TO7:E4$=E4$+" "+CHR$(RND(35)+129):NEXT:FORX1=1TO3:E5$=E5$+CHR$(RND(30)+157)+" ":
NEXTX1:CL$=STRING$(15,32)
320 BASE=475:TIME=60:S=0
330 CLS:SET(54,0):SET(126,3):SET(10,5):SET(94,8):SET(96,11):SET(76,25):SET(34,28
):SET(103,35):SET(2,28):SET(61,39)
340 PRINT@960,"Score ";S;STRING$(14,32);"Hiscore ";HS;STRING$(14,32);"Time Left"
;INT(TIME);:PRINT@BASE,FA$;:LBASE=BASE
350 Z=RND(21):ONZGOTO360,370,380,390,400,410,420,430,440,450,460,470,480,490,500
,510,520,530,540,550,560
360 A$=A1$:GOTO570:REM ALIEN 1 COMES ON SCREEN
370 A$=A2$:GOTO570:REM ALIEN 2 COMES ON SCREEN
```

Alien wipeout is another of the 'shoot-em-up' type games for the TRS80/SYSTEM 80. It requires approximately 7K and (with minor modifications) will run on both 16K and disk based machines, the listing here being the 16K version. The game itself is pretty self explanatory but here are a few hints which might make you a better player:

1. The aliens fuel store is located at the left uppermost point on the aliens ship and getting this point within your sights will certainly destroy the alien. If you hit the alien anywhere other than this point the impact will only cause minor damage.
2. 'Warping' from one side of the screen to the other disrupts your firing and your viewfinder. Although you might think you hit the alien when in this position it is quite possible that the unusual space time continuum of hyperspace will have absorbed your shot.
3. When the situation arises that the alien is on the opposite side of the screen don't waste your precious seconds by heading straight for him. Instead 'warp' from one side of the screen to the other and you will usually be able to catch him faster. Although this is a simple and quite obvious procedure I have included it among your hints for the simple reason that it is incredibly easy to forget in the heat of battle!

Finally for those of you who have disk based systems please note the following modifications to the program.

Line 80: Insert a CMD"T": in front of the CLEAR command.

Line 240: Replace the POKE16396,175 with CMD"BREAK.N" (or the equivalent command for your DOS)

Line 970: Replace the POKE16396,201:END with CMD"S" Line 990: Replace the POKE16526,62:POKE16527,64 with

DEFUSR0 = 16446

DEFUS

Lines 1010-1120: Replace all USR(0)'s with USR0(0)

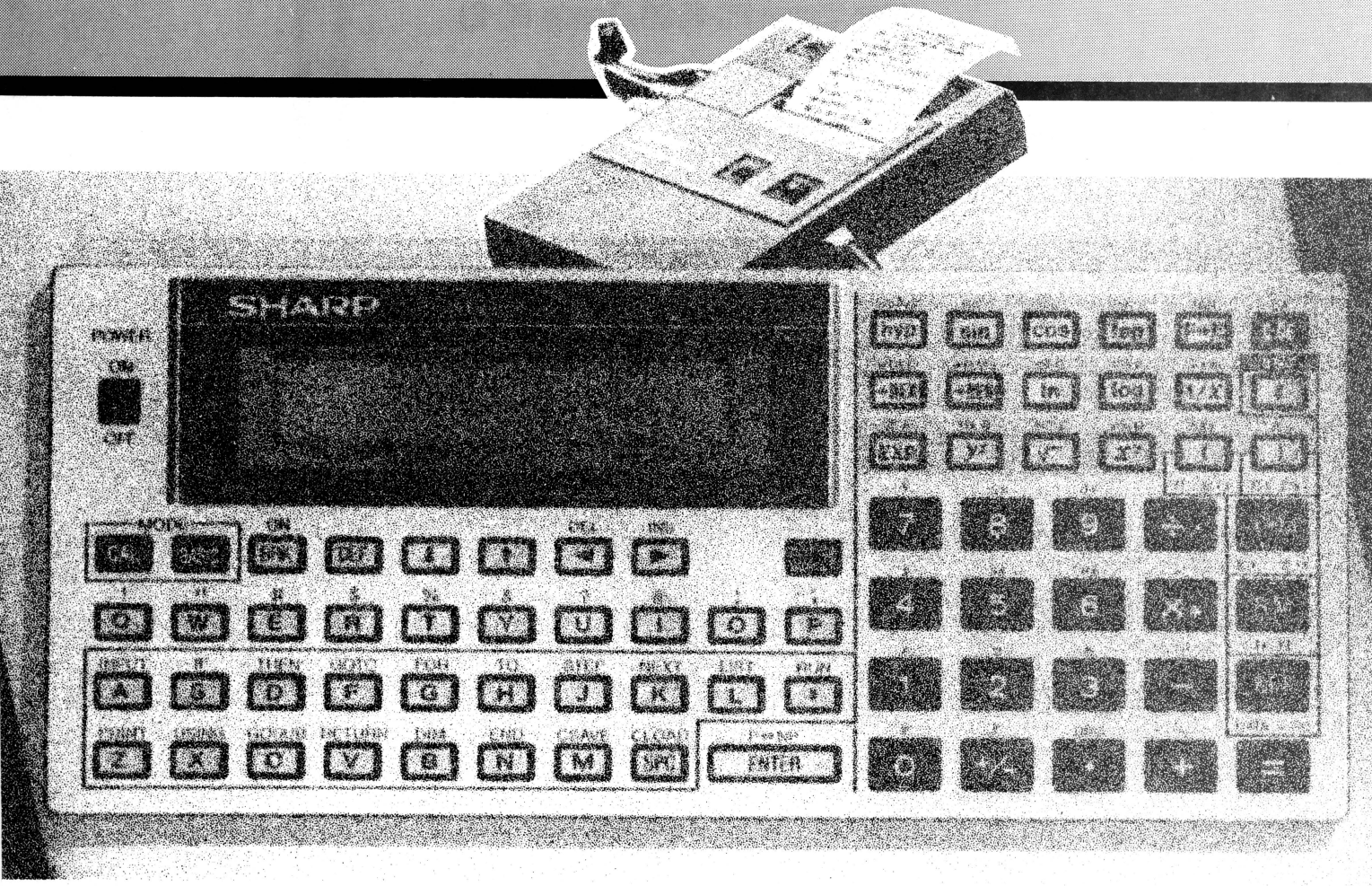
Wayne McCullough
Darwin NT

▽

ALIEN WIPEOUT

```
380 A$=A3$:GOTO570:REM ALIEN 3 COMES ON SCREEN
390 A$=A4$:GOTO570:REM ALIEN 4 COMES ON SCREEN
400 A$=A5$:GOTO570:REM ALIEN 5 COMES ON SCREEN
410 A$=A6$:GOTO570:REM ALIEN 6 COMES ON SCREEN
420 A$=A7$:GOTO570:REM ALIEN 7 COMES ON SCREEN
430 A$=A8$:GOTO570:REM ALIEN 8 COMES ON SCREEN
440 A$=A9$:GOTO570:REM ALIEN 9 COMES ON SCREEN
450 A$=B1$:GOTO570:REM ALIEN 10 COMES ON SCREEN
460 A$=B2$:GOTO570:REM ALIEN 11 COMES ON SCREEN
470 A$=B3$:GOTO570:REM ALIEN 12 COMES ON SCREEN
480 A$=B4$:GOTO570:REM ALIEN 13 COMES ON SCREEN
490 A$=B5$:GOTO570:REM ALIEN 14 COMES ON SCREEN
500 A$=B6$:GOTO570:REM ALIEN 15 COMES ON SCREEN
510 A$=B7$:GOTO570:REM ALIEN 16 COMES ON SCREEN
520 A$=B8$:GOTO570:REM ALIEN 17 COMES ON SCREEN
530 A$=B9$:GOTO570:REM ALIEN 18 COMES ON SCREEN
540 A$=C1$:GOTO570:REM ALIEN 19 COMES ON SCREEN
550 A$=C2$:GOTO570:REM ALIEN 20 COMES ON SCREEN
560 A$=C3$:REM ALIEN 21 COMES ON SCREEN
570 A=RND(888):B=A:PRINT@A,A$;
580 P=PEEK(14400):IFP=0THEN780:' GOTO MOVING ALIEN
590 IFP<>128THEN710:REM FIRE
600 PRINT@BASE,FB$;:PRINT@A,A$;:GOSUB1010:IFA>BASEANDA<BASE+70RA>BASE+64ANDA<BAS
E+71THEN620
610 FORI=1TO50:NEXTI:PRINT@BASE,FA$;:GOTO780:'GOTO MOVING ALIEN
620 P1=A-3:P2=A-67:P3=A+61:P4=A-128:P5=A+128:PRINT@P1,E4$;:IFP2>0ANDP2<1024THENP
RINT@P2,E3$;
630 IFP3>0ANDP3<1024THENPRINT@P3,E3$;
640 IFP4>0ANDP4<1024THENPRINT@P4,E5$;
650 IFP5>0ANDP5<1024THENPRINT@P5,E5$;
660 ONRND(3)GOTO670,680,690
670 GOSUB1110:GOTO700
680 GOSUB1020:GOTO700
690 GOSUB1040
700 S=S+1:IFS>HSTHENHS=S:GOTO330:ELSE330
710 IFP=8THENBASE=BASE-64:GOTO760
720 IFP=16THENBASE=BASE+64:GOTO760
730 IFP=32THENBASE=BASE-2:GOTO760
740 IFP=64THENBASE=BASE+2:GOTO760
750 GOTO780:'MOVING ALIEN
760 IFBASE>888ORBASE<0THENBASE=LBASE:GOTO780:'MOVING ALIEN
770 PRINT@LBASE,FC$;:PRINT@BASE,FA$;:LBASE=BASE
780 ' MOVING ALIEN
790 Z=RND(4):ON Z GOTO 800,810,820,830
800 A=A+64:GOTO840
810 A=A-64:GOTO840
820 A=A+2:GOTO840
830 A=A-2
840 IFA>888ORA<0THENA=B:GOTO790
850 PRINT@B,FC$;:PRINT@A,A$;:B=A:TIME=TIME-.4:IFTIME<=0THEN880
860 SET(54,0):SET(126,3):SET(10,5):SET(94,8):SET(96,11):SET(76,25):SET(34,28):SE
T(103,35):SET(2,28):SET(61,39)
870 PRINT@960,"Score ";S;STRING$(14,32);"Hiscore ";HS;STRING$(14,32);"Time Left"
;INT(TIME);:IFINT(TIME)<>OTTHENDV=USR(6404):OT=INT(TIME):GOTO580ELSEGOTO580
880 GOSUB1120:REM END
890 CLS:PRINT@23,"G a m e  O v e r":PRINT@87,"================":PRINT:PRINTTAB(2
4);"Your Score =";S:PRINTTAB(24);"Hi-Score =";HS;:PRINT:PRINT
900 IFS<5THENPRINT"The aliens have totally taken over the planet and you are a
   discrace to the armed forces of the Earth!":GOSUB1100:GOTO950
910 IFS<7THENPRINT"You managed to save one part of the Earth...Siberia!":GOSUB11
00:GOTO950
920 IFS<10THENPRINT"Not bad! You managed to save the entire southern hemisphere
fromthe aliens!":GOSUB1080:GOTO950
930 IFS<15THENPRINT"Very Good! You managed to repel the aliens from all of the
    planet except for a large island in the southern hemisphere     called Austr
alia!":GOSUB1030:GOTO950
940 IFS>=15THENPRINT"FANTASTIC! You have destroyed all the aliens and the human
race has a home once more!!!!":FORZZ=1TO2:GOSUB1110:NEXTZZ
950 PRINT:PRINT:PRINT"SIR!  Long range sensors report a wall of enemy craft head
ing   this way.  Do you wish to take comman of this mission?";CHR$(95);
960 A$=INKEY$:IFA$="Y"THEN320ELSEIFA$="y"THEN320ELSEIFA$="N"THEN970ELSEIFA$="n"T
HEN970ELSE960
970 CLS:POKE16396,201:END
980 REM ******* SOUND ROUTINES
990 FORC=16446TO16474:READD:POKEC,D:NEXTC:POKE16526,62:POKE16527,64:DATA205,127,
10,62,1,14,0,237,91,61,64,69,47,230,3,179,211,255,13,40,4,16,246,24,242,37,32,24
1,201
1000 RETURN
1010 FORC=1TO5:DV=USR(333):DV=USR(3000):DV=USR(333):NEXT:RETURN
1020 FORC=1TO5:DV=USR(2575):DV=USR(2560):FORD=1TO10:NEXT:NEXT:RETURN
1030 FORZZ=1TO5:FORC=5TO1STEP-1:DV=USR(3333/C):NEXTC,ZZ:RETURN
1040 FORC=200TO2000STEP100:DV=USR(C):NEXT:FORC=1TO3+RND(9):DV=USR(2580-C):FORSR=
1TO3:DV=USR(260):NEXT:NEXT:RETURN
1050 FORC=1TO3:GOSUB1070:NEXT:FORC=1TO3:DV=USR(12819):GOSUB1060:NEXT:FORC=1TO3:G
OSUB1070:NEXT:FORD=1TO100:NEXT:RETURN
1060 FORD=1TO25:NEXT:RETURN
1070 DV=USR(1299):GOSUB1060:RETURN
1080 DV=USR(7709):DV=USR(7707):DV=USR(7713):DV=USR(7723):DV=USR(10279):RETURN
1090 FORC=600TO520STEP-1:DV=USR(C):NEXT:FORC=520TO600:DV=USR(C):NEXT:RETURN
1100 FORC=1TO5:DV=USR(13055):FORD=1TO50:NEXT:NEXT:RETURN
1110 FORC=390TO260STEP-10:DV=USR(C):NEXT:FORC=260TO360:DV=USR(C):NEXT:RETURN
1120 FORC=1TO5:DV=USR(25675+C*30):GOSUB1060:NEXT:C=USR(25855):RETURN
1130 END
```

PROGRAMS FOR SHARP



TEXT EDITOR

(For Sharp PC-1500, Tandy PC-2)

All colours may be selected and provision for text input which exceeds the maximum line length has been made in the following manner:

Text is input at line 130 of the program and lines 150-190 test the length of variable A$(0) in relation to the size of lettering in use. They transfer control to line 300 if the text exceeds the maximum characters allowable for the selected size.

The program then requests a change of letter size, or amendment of text. It is then displayed on the screen, with the excess characters in brackets.

Lines 11 to 16 may be deleted when the user is familiar with the program use. The rotate command enables printing downwards and that section of the program may be removed if not required.

The program is extremely simple to use, and owners will need only spend a minimum of time to become familiar with it.

Robert Christian
Harbord NSW

```
 5:"TE"
 6:LOCK
 9:CLEAR
10:PAUSE "  ** T
   ext Editor **"
   :PAUSE "Copyr.
    R. Christian
   1983"
11:PRINT "(T)= Te
   xt"
12:PRINT "(C)= Co
   lour"
13:PRINT "(S)= Si
   ze"
14:PRINT "(R)= Ro
   tate (directio
   n)"
15:PRINT "(L)= Li
   ne Feed"
16:PRINT "(*)= Cu
   rrent Status"
20:DIM A$(0)*36
30:A$=INKEY$
40:WAIT 0:PRINT "
   (T)/(C)/(S)/(R
   )/(L)/(*)"
50:IF (A$="L")+(A
   $="T")+(A$="C"
   )+(A$="S")+(A$
   ="R")+(A$="*")
   <>1THEN 30
60:IF A$="T"THEN
   GOTO "T"
70:IF A$="S"THEN
   GOTO "S"
80:IF A$="C"THEN
   GOTO "C"
90:IF A$="R"THEN
   GOTO "R"
91:IF A$="L"THEN
   "L"
92:IF A$="*"THEN
   400
95:IF A$=""THEN 3
   0
100:"R":GRAPH :
   INPUT "Rotate?
    ";R:IF R>4
   THEN 100
101:ROTATE R:GOTO
   30
110:"S":TEXT :
   INPUT "Size? "
   ;S:IF S<1OR S>
   9THEN 110
111:CSIZE S:GOTO 3
   0
120:"C":TEXT :
   INPUT "Colour?
    ";C:IF C>4
   THEN 120
121:COLOR C:GOTO 3
   0
130:"T":GRAPH :
   COLOR C:CSIZE
   S:ROTATE R
140:A1=35:A2=18:A3
   =12:A4=9:A5=7:
   A6=6:A7=5:A8=4
145:INPUT "Text? "
   ;A$(0)
150:IF S=1AND LEN
   A$(0)>A1THEN 3
   00
155:IF S=2AND LEN
   A$(0)>A2THEN 3
   00
160:IF S=3AND LEN
   A$(0)>A3THEN 3
   00
170:IF S=4AND LEN
   A$(0)>A4THEN 3
   00
175:IF S=5AND LEN
   A$(0)>A5THEN 3
   00
180:IF S=6AND LEN
   A$(0)>A6THEN 3
   00
185:IF S=7AND LEN
   A$(0)>A7THEN 3
   00
190:IF S>=8AND LEN
   A$(0)>A8THEN 3
   00
191:IF S<1GOTO 425
200:INPUT "Number
   of doubles? ";
   D
210:FOR I=0TO D-1:
   GLCURSOR (I,I)
   :LPRINT A$(0):
   NEXT I
250:"L":TEXT :
   CSIZE S:INPUT
   "Line Feed? ";
   L
251:IF S<1THEN 425
252:LF L:GOTO 30
300:PAUSE "Delete
   Chr(s) in Brkt
   s."
301:PAUSE "or chan
   ge size"
302:WAIT :IF S=1
   PRINT "(";
   RIGHT$ (A$(0),
   LEN A$(0)-A1);
   ")":GOTO 30
305:IF S=2PRINT
   LEFT$ (A$(0),A
   2);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A2);")":
   GOTO 30
310:IF S=3PRINT
   LEFT$ (A$(0),A
   3);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A3);")":
   GOTO 30
315:IF S=4PRINT
   LEFT$ (A$(0),A
   4);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A4);")":
   GOTO 30
320:IF S=5PRINT
   LEFT$ (A$(0),A
   5);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A5);")":
   GOTO 30
325:IF S=6PRINT
   LEFT$ (A$(0),A
   6);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A6);")":
   GOTO 30
330:IF S=7PRINT
   LEFT$ (A$(0),A
   7);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A7);")":
   GOTO 30
335:IF S>=8PRINT
   LEFT$ (A$(0),A
   8);"(";RIGHT$
   (A$(0),LEN A$(
   0)-A8);")":
   GOTO 30
400:IF C=0THEN LET
   B$="BLK"
402:IF C=1THEN LET
   B$="BLU"
403:IF C=2THEN LET
   B$="GRN"
404:IF C=3THEN LET
   B$="RED"
405:WAIT
410:PRINT "C=";B$;
   "/S=";S;"/R=";
   R;"/LF=";L
420:GOTO 30
425:PAUSE "No Size
    Signal Given"
   :GOTO "S"


-----------------
Bytes used: 1709
```

PRIME NUMBERS

(For Sharp PC-1500)

After trying the Benchmark Prime Number test and recording an earthshattering 4520 sec on a Sharp 1500 I feel that I cannot seriously take up the speed challenge a *Your Computer* correspondent put out in a recent issue. However I am able to comment on improvements that enabled me to reduce the time taken to 84 sec.

My first thoughts were to improve the program along the following lines:- That trail divisors need only be primes greater than the square root of the number being tested.

That even numbers can be omitted from the routine altogether.

This reduced the time to a more respectable 432 sec.

At this point I remembered a procedure usually taught to year 7 pupils called the Sieve of Eratosthenes (also a major benchmark for computers,Ed.). The enclosed listing is the result. If a similar improvement is possible for other machines then I would expect to see a time of 1 sec.

The program sets up a matrix of 1000 elements, then sets up a table of primes greater than 32. Odd composite numbers are then eliminated by repeated addition. Printing is carried out in a loop. After all, the challenge did not say that you had to be able to read them!

Ian McIntyre
Mooroolbark VIC

```
$:"A"CLEAR :WAIT
  0:DIM D$(3,249
  )*1,D(255)
10:T=TIME
35:FOR N=3TO 31
  STEP 2
40:IF N<2THEN 60
45:FOR K=2TO N
50:IF N/K=INT (N/
  K)THEN 65
55:NEXT K
60:D(A)=N,A=A+1
65:NEXT N
70:FOR I=0TO A-1
75:FOR J=D(I)*D(I
  )TO 1000STEP D
  (I)*2
80:X=INT ((J-1)/2
  50),Y=J-250*X-
  1
85:D$(X,Y)="*"
90:NEXT J
95:NEXT I
100:PRINT "2":D(0)
  =2,L=1
105:FOR I=0TO 3
110:FOR J=0TO 249
  STEP 2
115:IF D$(I,J)="*"
  THEN 125
120:D(L)=250*I+J+1
  ,L=L+1:PRINT D
  (L-1)
125:NEXT J
130:NEXT I

135:S=TIME
140:WAIT :PRINT L,
  INT ((DEG S-
  DEG T)*3600)
145:END
```

MACHINE CODE MONITOR

(For Sharp PC-1500)

Minimum System: PC-1500 with 8K RAM and cassette interface/ printer.
Recommended Reference: Sharp PC-1500 Pocket Computer Technical Reference Manual.

Pocket Monitor is a basic machine code monitor for the Sharp PC-1500 computer. It includes facilities to examine and/or change memory contents, printout the contents of specific areas of memory, run machine code programs and set or examine some of the CPU registers before, or after, running a machine code program.

The program is controlled by selecting the required function from a six item menu using the function keys.

S. Corrigan
Lawson NSW

```
  10 "M"PAUSE "MONITOR.1 9/10/83"
  20 REM  INITIALISATION
  30 GOSUB "IN"
  40 REM  COMMAND
  50 "CM"BEEP 2:WAIT 0:PRINT "  ME  PR
RE  GO  MO  HELP"
  60 CM=ASC INKEY$ -16:IF CM<10R CM>=7T
HEN 60
  70 WAIT
  80 ON CMGOSUB "ME","PR","RE","GO","MO
","HE"
  90 GOTO "CM"
 100 "ME"REM  MEMORY
 110 INPUT "ADDRESS:";AD
 120 IF I$="0"LET DA=PEEK AD:GOTO 125
 122 DA=PEEK# AD
 125 IF M$="D"WAIT 0:PRINT "ME";I$;AD;"
:";DA;" (H for help)":GOTO 160
 130 J=DA:GOSUB "HEX2":DA$=F$
 140 J=AD:GOSUB "HEX4":AD$=F$
 150 WAIT 0:PRINT "ME";I$;" &";AD$;":&"
;DA$;" (H for help)"
 160 ST=ASC INKEY$
 170 IF ST=10OR ST=13THEN "NX"
 180 IF ST=11THEN "LS"
 190 IF ST=67THEN "WR"
 195 IF ST=72THEN "MH"
 200 IF ST=32THEN RETURN
 210 GOTO 160
 220 "NX"AD=AD+1:GOTO 120
 230 "LS"AD=AD-1:GOTO 120
 240 "WR"INPUT "DATA:";DA
 250 IF I$="0"POKE AD,DA:GOTO "NX"
 260 POKE# AD,DA:GOTO "NX"
 265 "MH"WAIT :GOSUB "HE":GOTO "ME"
 270 "PR"REM  PRINT
 280 INPUT "FROM:";SA
 290 INPUT "TO:";EA
 292 LPRINT "* DATA FROM ME";I$;" *"
 293 LPRINT "-----------------"
 295 IF M$="H"THEN 330
 298 AD=SA
 300 FOR Z=0TO 3:IF I$="0"LET D(Z)=PEEK
 AD:GOTO 310
 305 D(Z)=PEEK# AD
 310 AD=AD+1:NEXT Z
 315 USING :LPRINT SA;":-"
 320 USING "####":LPRINT " ";D(0);D(1);
D(2);D(3)
 322 SA=SA+4
```

MACHINE CODE MONITOR

```
  325 IF SA>=EATHEN USING :RETURN
  327 GOTO 300
  330 J=SA:GOSUB "HEX4":SA$=F$
  335 FOR Z=0TO 3
  340 IF I$="0"LET D(Z)=PEEK SA:GOTO 350
  345 D(Z)=PEEK# SA
  350 J=D(Z):GOSUB "HEX2":D$(Z)=F$
  360 SA=SA+1
  370 NEXT Z
  375 LPRINT "&";SA$;":-"
  380 LPRINT "   &";D$(0);",&";D$(1);",&
",D$(2);",&";D$(3)
  390 IF SA>=EATHEN RETURN
  395 GOTO 330
  400 "GO"
  410 POKE &57E1,&A5,&57,&A5,&0A,&A5,&57
,&A6,&08
  420 POKE &57E9,&A5,&57,&A7,&BE,&57,&FF
,&AE,&57
  430 POKE &57F1,&A7,&FD,&AA,&AE,&57,&A8
,&04,&AE
  440 POKE &57F9,&57,&A5,&84,&AE,&57,&A6
,&9A
  450 INPUT "START ADDRESS: ";SA
  460 SH=INT (SA/256):SL=SA-INT (SH*256)

  470 POKE &57ED,SH,SL:REM  SET SJP
  480 CALL &57E1
  490 RETURN
  500 "MO"REM  SET DISPLAY
  510 WAIT 0:PRINT "DISPLAY (H)EX OR (D)
EC?"
  520 M$=INKEY$
  530 IF M$="H"PAUSE "HEX DISPLAY":GOTO
560
  540 IF M$="D"PAUSE "DECIMAL DISPLAY":G
OTO 560
  550 GOTO 510
  560 PRINT "ME area, (0) or (1)?"
  570 I$=INKEY$
  580 IF I$="1"OR I$="0"PAUSE "MEMORY AR
EA=ME";I$:RETURN
  590 GOTO 570
  600 "RE"REM  REGISTERS
  610 WAIT 0:PRINT "(S)ET or (R)EAD REGI
STERS?"
  620 R$=INKEY$
  630 IF R$="S"WAIT :GOTO 740
  640 IF R$<>"R"THEN 620
  645 WAIT
  650 XL=PEEK &57A5:XH=PEEK &57A6
  660 AC=PEEK &57A7:ST=PEEK &57A8
  670 IF M$="H"THEN 700
  680 PRINT "XH=";XH;" XL=";XL
  690 PRINT "ACC.=";AC;" STATUS=";ST:RET
URN
  700 J=XH:GOSUB "HEX2":XH$=F$:J=XL:GOSU
B "HEX2":XL$=F$
  710 J=AC:GOSUB "HEX2":AC$=F$:J=ST:GOSU
B "HEX2":ST$=F$
  720 PRINT "XH=&";XH$;" XL=&";XL$
  730 PRINT "ACC.=&";AC$;" STATUS=&";ST$
:RETURN
  740 INPUT "XH=";XH:INPUT "XL=";XL
  750 INPUT "ACC.=";AC
  760 POKE &57A5,XL,XH,AC
  770 RETURN
  800 "HE"PRINT "Step help menu with ENT
ER"
  810 BEEP 1:PRINT "  ME <-- memory.....
..."
  830 PRINT "...step addr: down arrow"
  840 PRINT "...back up addr: up arrow"
  850 PRINT "...to change, Press C"
  860 PRINT "....Press SPACE to exit."
  865 IF ST=72THEN RETURN
  870 BEEP 1:PRINT "       PR <-- Print m
emory."
  880 BEEP 1:PRINT "          RE <--regi
sters"
  890 PRINT "..set or read ACC. & X...."

  895 PRINT "..before or after GO."
  900 BEEP 1:PRINT "run Prog.---> GO"
  910 BEEP 1:PRINT "mode: dec/hex --->MO
"
  920 PRINT "...also sets ME0 or ME1."
  930 RETURN
 1000 "HEX4"F$="0000"
 1010 IF J>32767LET J=J-65536
 1020 CALL &57D8,J
 1030 CALL &57B4
 1040 CALL &57A9
 1050 RETURN
 1060 "HEX2"F$="00"
 1070 POKE &57A3,J
 1080 CALL &57B4
 1090 RETURN
 1100 "IN"REM  INITIALISE
 1110 REM  MACHINE CODE, CONVERT HEX TO
STRING
 1120 POKE &57A9,&A5,&57,&A4,&AE,&57,&A3
,&4A,&62
 1130 POKE &57B1,&BA,&57,&B6,&4A,&60,&40
,&76
 1140 POKE &57B8,&A5,&57,&A3,&D5,&D5,&D5
,&D5,&BE
 1150 POKE &57C0,&57,&CE,&0E,&44,&A5,&57
,&A3,&B9
 1160 POKE &57C8,&0F,&BE,&57,&CE,&0E,&9A
,&B7,&0A
 1170 POKE &57D0,&83,&03,&B3,&30,&9A,&B3
,&36,&9A
 1180 POKE &57D8,&84,&AE,&57,&A3,&04,&AE
,&57,&A4,&9A
 1190 CLEAR :DIM D(4):DIM D$(4)
 1200 M$="H":I$="0"
 1210 RETURN
```

PRINTOUT 1.

```
* DATA FROM ME0 *
-----------------
&57A9:-
   &A5,&57,&A4,&AE
&57AD:-
   &57,&A3,&4A,&62
&57B1:-
   &BA,&57,&B6,&4A
&57B5:-
   &60,&48,&76,&A5
&57B9:-
   &57,&A3,&D5,&D5
&57BD:-
   &D5,&D5,&BE,&57
&57C1:-
   &CE,&0E,&44,&A5
&57C5:-
   &57,&A3,&B9,&0F
&57C9:-
   &BE,&57,&CE,&0E
&57CD:-
   &9A,&B7,&0A,&83
&57D1:-
   &03,&B3,&30,&9A
&57D5:-
   &B3,&36,&9A,&84
&57D9:-
   &AE,&57,&A3,&04
&57DD:-
   &AE,&57,&A4,&9A
&57E1:-
   &A5,&57,&A5,&0A
&57E5:-
   &A5,&57,&A6,&08
&57E9:-
   &A5,&57,&A7,&BE
&57ED:-
   &57,&FF,&AE,&57
&57F1:-
   &A7,&FD,&AA,&AE
&57F5:-
   &57,&A8,&04,&AE
&57F9:-
   &57,&A5,&84,&AE
&57FD:-
   &57,&A6,&9A,&00
```

PRINTOUT 2.

```
* DATA FROM ME0 *
-----------------
22441:-
 165  87 164 174
22445:-
  87 163  74  98
22449:-
 186  87 182  74
22453:-
  96  72 118 165
22457:-
  87 163 213 213
22461:-
 213 213 190  87
22465:-
 206  14  68 165
22469:-
  87 163 185  15
22473:-
 190  87 206  14
22477:-
 154 183  10 131
22481:-
   3 179  48 154
22485:-
 179  54 154 132
22489:-
 174  87 163   4
22493:-
 174  87 164 154
22497:-
 165  87 165  10
22501:-
 165  87 166   8
22505:-
 165  87 167 190
22509:-
  87 255 174  87
22513:-
 167 253 170 174
22517:-
  87 168   4 174
22521:-
  87 165 132 174
22525:-
  87 166 154   0
```

MACHINE CODE SUB-ROUTINES

```
** CONVERT HEX NUMBER TO STRING IN F$
&57A9 &A557A4      C2>LDA (&57A4)
&57AC &AE57A3         STA (&57A3)
&57AF &4A62           LDI XL &62              ADDRESS F$
&57B1 &BA57B6         JMP BT
&57B4 &4A60       C1>LDI XL &60               ADDRESS F&
&57B6 &4876       BT>LDI XH &76               "   "   "
&57B8 &A557A3         LDA (&57A3)             BYTE TO BE ...
&57BB &D5             SHR                     CONVERTED,
&57BC &D5             SHR                     LEFT NIBBLE
&57BD &D5             SHR
&57BE &D5             SHR
&57BF &BE57CE         SJP AS                  CONV. TO ASCII
&57C2 &0E             STA (X)                 STORE IN F$
&57C3 &44             INC X
&57C4 &A557A3         LDA (&57A3)             BYTE TO BE....
&57C7 &B90F           ANI &0F                 CONVERTED, RIGHT
&57C9 &BE57CE         SJP AS
&57CC &0E             STA (X)                 STORE IN F$
&57CD &9A             RTN
&57CE &B70A       AS>CPI &0A                  >&0A?
&57D0 &8303           BCS LT                  YES
&57D2 &B330           ADI A &30               0 TO 9
&57D4 &9A             RTN
&57D5 &B336       LT>ADI A &36                A TO F
&57D7 &9A             RTN
    ROUTINE TO SAVE VARIABLE IN MEMORY
&57D8 &84             LDA XH
&57D9 &AE57A3         STA (&57A3)
&57DC &04             LDA XL
&57DD &AE57A4         STA (&57A4)
&57E0 &9A             RTN
     ROUTINE TO FETCH AND SAVE ACC. X REG. AND
          STATUS FOR GO FUNCTION
&57E1 &A557A5         LDA (&57A5)
&57E4 &0A             STA XL
&57E5 &A557A6         LDA (&57A6)
&57E8 &08             STA XH
&57E9 &A557A7         LDA (&57A7)
&57EC &BE57FF         SJP EN                  CALL USER.....
                   ** WARNING **
     SUBROUTINE JUMP ADDRESS SET BY BASIC AT LINE 470
&57EF &AE57A7         STA (&57A7)
&57F2 &FDAA           TTA                      ACC.=STATUS
&57F4 &AE57A8         STA (&57A8)
&57F7 &04             LDA XL
&57F8 &AE57A5         STA (&57A5)
&57FB &84             LDA XH
&57FC &AE57A6         STA (&57A6)
&57FF &9A             RTN
```

FRUIT MACHINE

(For Sharp MZ-700)

Here is a good program for beginners like me, because it incorporates sound and colour and is a lot of fun to play.

Allan Moss
Duffy ACT

```
THIS IS A $.50 SLOT MACHINE.
PAYOFF IS $3.00 FOR 3 CHERRIES, 3 LEMONS
, OR 3 ORANGES.
ALL OTHER COMBINATIONS LOSE.
HOW MANY 50-CENT PIECES DO YOU WANT TO U
SE IN PLAY.
? 4
YOU START WITH $ 2

DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
   LEMON       LEMON       LEMON
GREAT YOU WON $3.
YOU NOW HAVE $ 5
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
  CHERRY      CHERRY      CHERRY
GREAT YOU WON $3.
YOU NOW HAVE $ 8
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
  CHERRY       ORANGE       ORANGE
          TO BAD - - YOU LOST $.50.


YOU NOW HAVE $ 7.5
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
   ORANGE       ORANGE      CHERRY
          TO BAD - - YOU LOST $.50.


YOU NOW HAVE $ 7
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
   LEMON       ORANGE      CHERRY
          TO BAD - - YOU LOST $.50


YOU NOW HAVE $ 6.5
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
   ORANGE       LEMON       LEMON
          TO BAD - - YOU LOST $.50.


YOU NOW HAVE $ 6
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 1
  CHERRY       ORANGE       LEMON
          TO BAD - - YOU LOST $.50.


YOU NOW HAVE $ 5.5
DO YOU WISH TO PLAY (TYPE 1 FOR YES, 0 F
OR NO)? 0
SORRY ABOUT THAT
Ready
```

```
6 PRINT ,"FRUIT MACHINE GAME FOR
         SHARP MZ-700 COMPUTER"
7 PRINT
8 PRINT
9 PRINT [, 4]
10 PRINT "THIS IS A $.50 SLOT MACHINE."
20 PRINT "PAYOFF IS $3.00 FOR 3 CHERRIES
, 3 LEMONS, OR 3 ORANGES."
30 PRINT "ALL OTHER COMBINATIONS LOSE."
40 PRINT "HOW MANY 50-CENT PIECES DO YOU
 WANT TO USE IN PLAY."
50 INPUT M
60 LET M=M*.5
70 PRINT "YOU START WITH $";M
80 LET X=RND(-1)
81 PRINT
90 PRINT "DO YOU WISH TO PLAY (TYPE 1 FO
R YES, 0 FOR NO)";
100 INPUT A
110 IF A=0 THEN 410
120 LET C=0
130 LET L=0
140 LET O1=0
150 FOR I=1 TO 3
160 LET N=INT(3*RND(1))+1
170 ON N GOTO 180,210,240
180 PRINT [7,0] "  CHERRY   ";
190 LET C=C+1
200 GOTO 260
210 PRINT [7, 2]"   LEMON   ";
220 LET L=L+1
230 GOTO 260
240 PRINT [7, 3] "   ORANGE   ";
250 LET O1=O1+1
260 NEXT I
270 IF C=3 THEN 350
280 IF L=3 THEN 350
290 IF O1=3 THEN 350
291 PRINT
300 PRINT ,"TO BAD - - YOU LOST $.50.";
301 MUSIC "#A1"
310 LET M=M- .5
311 PRINT
320 PRINT
330 IFM=0 THEN 400
331 PRINT
340 GOTO 380
350 PRINT ,"GREAT YOU WON $3.";
355 MM$= "C1+D1+E1+F1+G1+A1+B1"
356 TEMPO 7
357 MUSIC MM$,MM$,MM$
360 LET M=M+3
370 PRINT
380 PRINT "YOU NOW HAVE $";M
390 GOTO 90
391 PRINT
400 PRINT "YOU HAVE LOST ALL YOUR MONEY.
"
401 PRINT
410 PRINT "SORRY ABOUT THAT"
411 MUSIC "-B4R+-5BR+-C8"
420 END
```

FIND THE PEA

(For Sharp PC-1500)

An age old carnival con game that has found its way onto a computer. Full instructions are in the program. Good luck!

Alan Thomas
Napier NZ

```
 1:REM  FIND THE
       PEA!
 2:REM    by
    ALLAN THOMAS
 3:REM 1.4.84!
10:CLEAR :RANDOM
   :B$="00081C080
   0"
20:FOR I=1TO 3
30:GOSUB 500
60:PAUSE " "
70:WAIT :CLS :
   PAUSE "FIND TH
   E PEA!"
80:NEXT I
90:INPUT "RULES=R
   /PLAY=P:? ";A$
100:IF LEFT$ (A$,1
   )="E"END
110:IF LEFT$ (A$,1
   )="R"GOTO 200
120:GOTO 300
200:BEEP 5:DIM Q$(
   0)*26:WAIT
210:RESTORE :FOR I
   =0TO 12
220:READ Q$(0):
   PRINT Q$(0)
230:NEXT I
240:GOTO 90
300:CLS :GOSUB 500
   :BEEP J*3,RND
   3*10:CURSOR 20
   :PRINT "?"
310:A$=INKEY$ :IF
   A$=""GOTO 310
320:KEY=ASC A$:IF
   KEY<17OR KEY>1
   9GOTO 310
330:N=N+1:P=RND 3:
   KE=KE-16
340:CLS :GCURSOR 2
   4*(P-1)+12:
   GPRINT B$
350:IF KE=PGOTO 40
   0
360:PAUSE "YOU LOS
   E!":PZ=-20:
   GOSUB 550:GOTO
   425
400:PZ=400-100*N:
   IF PZ<0LET PZ=
   0
410:PAUSE "YOU WIN
   $";PZ;" !!":
   GOSUB 550:N=0:
   GOTO 300
425:IF N<3GOTO 300
430:N=0:INPUT "AGA
   IN? ";Q$:IF
   LEFT$ (Q$,1)="
   Y"GOTO 300
440:END
500:WAIT 50:CLS
510:FOR J=1TO 3
520:BEEP J:CURSOR
   4*J-2:PRINT
   CHR$ 127;
530:NEXT J:RETURN
550:TP=TP+PZ:PAUSE
   "TOTAL WINNING
   S $";TP:RETURN
600:DATA "RULES:Pr
   ess Enter for
   more.","3 cups
    are on the ta
   ble."
610:DATA "The pea
   is under one c
   up.","Which cu
   p?","You are a
   llowed 3 guess
   es."
630:DATA "The cups
    are shuffled"
   ,"after each g
   uess","Press k
   ey under cup c
   hosen"
640:DATA "Prizes f
   or finding Pea
   :","First try:
    $300","Second
    try: $200"
650:DATA "Third tr
   y: $100","Each
    wrong guess:
   -$20"


STATUS 1
                1060
```

DECIMAL TO HEX

(For Sharp PC-1500)

Simply put in a number when asked and the hexadecimal value will be calculated and displayed.

Alan Thomas
Napier NZ

```
10:"DH"CLEAR :H$=
   "0123456789ABC
   DEF"
30:INPUT "Number=
   ";N:IF N>65535
   PRINT "ABOVE L
   IMIT!":GOTO 30
35:IF N<0END
40:FOR I=1TO 4
50:GOSUB 100
60:NEXT I
70:N$="&"+N$
80:PRINT N$
90:N$="":GOTO 30
100:A=INT (N/16):Z
   =N-16*A:N$=
   MID$ (H$,(Z+1)
   ,1)+N$:N=A:
   RETURN
```

ESCAPE

(For Sharp PC-1401)

To play ESCAPE just type RUN and press ENTER. A prompt will ask you for the size of the Maze. The computer will then be busy while it generates the new Maze. The computer will beep twice when the maze is completed.

During play the computer will display the available exits open to you, these will be any combination of N(orth), E(ast), W(est), or S(outh) inside square brackets. Just press the key of the compass point of the direction you wish to move.

The computer can generate a maze with one-way walls and rooms with no exits.

'YOU ARE TRAPPED!' is displayed when in a room of no exits, pressing 'X' will take you back to the start of the maze.

'FREE AT LAST' and your score (maximum is 1000) will be displayed when you find your way out of the maze.

David Green
Shailer Park QLD

```
10:CLEAR :PAUSE "** ESCAPE! **":WAIT 25
   :X=0:Y=0:S$="*"
20:RANDOM :INPUT "MAZE SIZE ";Z:Z=Z-1:
   DIM N$(Z,Z)*1,E$(Z,Z)*1,W$(Z,Z)*1,
   S$(Z,Z)*1:T=1001+Z
30:D=RND (Z/2):C=RND 4
40:FOR L=1 TO D:ON C GOTO 50,55,50,55
50:N$(X,Y)=S$:S$(X,Y)=S$:GOTO 60
55:E$(X,Y)=S$:W$(X,Y)=S$
60:X=X+(C=2)-(C=4):Y=Y+(C=3)-(C=1)
70:IF X 0 LET X=0:L=D
80:IF X Z LET X=Z:L=D
90:IF Y 0 LET Y=0:L=D
100:IF Y=Z+1 THEN 120
110:NEXT L:GOTO 30
120:FOR L=0 TO Z:W$(0,L)="":S$(L,0)="":
   E$(Z,L)="":NEXT L:BEEP 2
130:X=0:Y=0
140:DD$="EXITS "+CHR$ 91:IF N$(X,Y)=S$
   LET DD$=DD$+"N"
141:IF E$(X,Y)=S$ LET DD$=DD$+"E"
142:IF W$(X,Y)=S$ LET DD$=DD$+"W"
143:IF S$(X,Y)=S$ LET DD$=DD$+"S"
144:DD$=DD$+CHR$ 93:IF LEN DD$=8 LET
   DD$="YOU ARE TRAPPED!"
150:PRINT DD$:O$=INKEY$ :IF O$=""
   THEN 150
155:IF O$="X" THEN 130
156:T=T-1
160:IF O$="N" OR O$="S" LET Y=Y+(O$="N"
   AND N$(X,Y)=S$)-(O$="S" AND S$(X,Y)
   =S$):GOTO 169
165:X=X+(O$="E" AND E$(X,Y)=S$)-(O$="W"
   AND W$(X,Y)=S$)
169:IF Y =Z THEN 140
170:WAIT :PRINT "FREE AT LAST":PRINT
   "SCORE :";T:END
```

PROGRAMS FOR VZ200

GOLF SIMULATION

This draws a golf course in graphics mode with endless variations on bunkers, water hazards and roughs, and allows the player to actually 'play' the shots giving a choice of club, hitting strength and direction.

Gary McCleary
Emu Plains NSW

```
40 REM GOLF SIMULATION
50 REM BY GARY J MCCLEARY
51 REM DEC. 1983
100 CLS
110 PRINT@33,"WELCOME TO GLENLAY GOLF CO
URSE"
111 PRINT
112 PRINT"IN GOLF THE OBJECT OF THE GAME
"
113 PRINT"IS TO HIT THE BALL FROM THE"
114 PRINT"TEE(O) TO THE HOLE IN THE"
115 PRINT"FEWEST NUMBER OF SHOTS."
120 PRINT
125 PRINT"WILL THERE BE 1 OR 2 PLAYERS?"
130 K$=INKEY$
133 I$=INKEY$:WW=RND(DD):DD=DD+1:IFDD>10
0THENDD=1:IFI$=""THEN133
135 IFI$="1"THENPL=1:LP=0:GOTO145
137 IFI$="2"THENPL=2:LP=0:GOTO145
140 GOTO130
145 CLS
155 PRINT"YOUR GOLF BAG CONTAINS A:"
158 PRINT
160 PRINT"1 WOOD MAX.RANGE 251 METRES"
165 PRINT"2 IRON MAX.RANGE 221 METRES"
170 PRINT"5 IRON MAX.RANGE 164 METRES"
175 PRINT"7 IRON MAX.RANGE 127 METRES"
180 PRINT"9 WEDGE MAX RANGE 87 METRES"
185 PRINT"PUTTER MAX.RANGE 41 METRES"
190 PRINT"AND IS ONLY USED ON THE GREEN"
194 PRINT
195 PRINT"TO ACHIEVE GREATER HEIGHT"
200 PRINT"USE A HIGH NUMBERED IRON"
205 PRINT
210 PRINT"SPACE CONTINUES THE GAME"
250 GOSUB20980
300 HO=1:TT=0:T1=0:T2=0:GF=0
350 PA=RND(3)+2
351 PZ=RND(2)
352 IFPA=3THENP=3:SX=63:GOTO400
354 IFPA=4THENP=4.8
356 IFPA=5THENP=6.5
368 IFPZ=1THENSX=8
370 IFPZ=2THENSX=119
400 REM
420 ZB=RND(3):ZW=RND(3):ZJ=RND(3)
430 J3=RND(9)+2
450 A=RND(107)+7:BB=RND(7)+16
453 G=RND(5)+2:B=RND(9)+2:W=RND(10)+3
455 IFZJ=1THENJ3=0
456 IFZB=1THENB=0
457 IFZW=1THENW=0
458 C=RND(103)+9:D=13+RND(6)
459 MD=INT(SQR((A-SX)^2+(BB-63)^2)*P)
460 HB=SQR((A-C)^2+(BB-D)^2)
465 IFHB<=G+B+3THEN458
466 E=13+RND(100):F=14+RND(35)
468 BW=SQR((C-E)^2+(D-F)^2)
470 WH=SQR((A-E)^2+(BB-F)^2)
472 IFBW<=B+W+3THEN466
474 IFWH<=W+G+3THEN466
480 J1=RND(103)+9:J2=RND(6)+13
485 HJ=SQR((A-J1)^2+(BB-J2)^2)
490 IFHJ<=G+J3+3THEN458
492 JW=SQR((J1-E)^2+(J2-F)^2)
494 IFJW<=J3+W+3THEN466
500 CLS
506 X=SX:Y=63:R1=0:B1=0:W1=0
507 SC=0
509 CLS
510 PRINT"THIS IS HOLE NUMBER" HO
511 PRINT
512 PRINT"PLAYER" LP+1
513 PRINT
514 PRINT"PAR"PA;  MD "METRES"
515 SC=0:X=SX:Y=63:R1=0:B1=0:W1=0
517 GOSUB20980
522 GOSUB20000
523 GOSUB20980
524 CLS
525 PRINT"WHICH CLUB DO YOU WISH TO USE"
527 INPUTCL
530 IFCL=1THENAV=29+RND(11):GOTO600
540 IFCL=2THENAV=19+RND(11):GOTO600
550 IFCL=5THENAV=69+RND(6):GOTO600
560 IFCL=7THENAV=74+RND(6):GOTO600
570 IFCL=9THENAV=79+RND(6):GOTO600
580 CLS:PRINT"YOU DO NOT HAVE ONE OF THO
SE":GOTO525
600 CLS
602 PRINT"IN WHICH DIRECTION DO YOU WISH
"
610 PRINT"TO HIT? (0TO360 DEGREES)"
620 PRINT"MEASURED ANTICLOCKWISE FROM"
630 PRINT"THE RIGHT"
635 GOSUB60300
640 INPUTAZ
645 CLS
650 PRINT"HOW HARD DO YOU WISH TO HIT"
660 INPUT"0TO50";U
665 CLS
668 PS=3.141592654/180
670 IFU<0THENU=0
675 IFU>50THENU=50
677 SC=SC+1
680 RA=U*U*SIN(2*AV*PS)/9.81
682 RS=RA/P
685 HT=((SIN(AV*PS)*U)^2)/(19.62)
686 IFR1=1THEN12000
687 IFB1=1THEN13000
690 X=X+RS*COS(AZ*PS)
700 Y=Y-RS*SIN(AZ*PS)
710 H=INT(X):K=INT(Y)
715 H1=0
720 IFH<0THENH=0:H1=1
725 IFH>=127THENH=126:H1=1
730 IFK<0THENK=0:H1=1
735 IFK>63THENK=63:H1=0
736 X=H:Y=K
740 IFH1=1THEN9000
742 DI=SQR((A-H)^2+(BB-K)^2)
745 REM
746 IFDI<=GANDGF=1THEN790
747 GOSUB20000
754 COLOR2
755 K$=INKEY$
760 I$=INKEY$
765 SET(H,K):SET(H+1,K)
770 RESET(H,K):RESET(H+1,K)
775 IFI$=""THEN760
780 IFI$<>" "THEN760
790 DI=SQR((A-H)^2+(BB-K)^2)
792 DB=SQR((C-H)^2+(D-K)^2)
794 DW=SQR((E-H)^2+(F-K)^2)
796 DJ=SQR((J1-H)^2+(J2-K)^2)
800 DM=DI*P
810 IFDI<=GTHENGF=1:GOTO8000
812 IFDB<=BANDB<>0THEN7000
813 IFDJ<=J3ANDJ3<>0THEN7000
814 IFDW<=WANDW<>0THEN10000
816 CLS
817 PRINT"THAT SHOT WENT "INT(RA)"METRES
"
819 PRINT
820 PRINT"DISTANCE FROM THE HOLE"
822 PRINTINT(DM)"METRES"
825 PRINT"NUMBER OF STROKES="SC
827 IFPA=4ORPA=5THEN1000
830 IFH<40ANDK>31THEN11000
835 IFH>86ANDK>31THEN11000
840 IFK<=8THEN11000
845 GOTO2000
1000 IFPZ=2THEN1500
1100 IFH>16ANDK>31THEN11000
1110 IFK<=8THEN11000
1120 GOTO2000
1500 IFH<111ANDK>31THEN11000
1510 IFK<=8THEN11000
1520 GOTO2000
2000 GOTO525
7000 B1=1
7005 BH=124.5
7010 PRINT"YOU ARE IN THE BUNKER"
7020 PRINT"YOU ARE ADVISED TO USE THE WE
DGE"
7030 GOTO525
8000 GF=1:GOTO60060
8004 CLS
8008 PRINT"YOU ARE ON THE GREEN AND WILL
"
8010 PRINT"BE USING THE PUTTER"
8020 PRINT"WHICH DIRECTION (0TO360)"
8025 GOSUB60300
8030 INPUTAZ
8035 CLS
```

VZ200

```
8040 PRINT"HOW HARD DO YOU WANT TO HIT"
8050 INPUT"(0TO25)";V
8060 IFV<0THENV=0
8065 IFV>25THENV=25
8070 AV=70
8075 CLS
8200 GOTO677
9000 SOUND4,2:SC=SC+1:GOTO745
10000 W1=0
10005 SC=SC+1
10010 H=H+2*W:K=K+2*W
10020 GOTO60000
11000 R1=1
11005 RH=111+RND(15)
11010 PRINT
11011 PRINT"YOU ARE IN THE ROUGH"
11012 IFRH>123THENB$="TALL TREES":GOTO11
018
11014 IFRH>118THENB$="MEDIUM TREES":GOTO
11018
11016 IFRH>=112THENB$="LOW SCRUB":GOTO11
018
11018 PRINT"YOUR NEXT SHOT MUST CLEAR SO
ME"
11019 PRINTB$
11020 PRINT
11030 GOTO525
12000 IFHT<RHTHENRA=RND(6):GOTO12100
12010 RA=RA/2
12100 R1=0:GOTO682
13000 IFHT<BHTHENRA=0:GOTO13100
13010 RA=RA/2
13100 B1=0:GOTO682
15000 SOUND20,1:SOUND15,1
15002 IFLP=0THENT1=T1+SC:TT=T1:P1=P1+SC-
PA:Q=P1
15003 IFLP=1THENT2=T2+SC:TT=T2:P2=P2+SC-
PA:Q=P2
15005 A$=" FOR THIS HOLE"
15008 CLS
15010 PRINT@39,"CONGRATULATIONS"
15015 PRINT@73,"PLAYER"LP+1
15020 PRINT
15030 PRINT"YOU ARE IN THE HOLE"
15040 PRINT"FOR "SC" SHOTS"
15060 IFSC=PA-2THENPRINT"EAGLE";A$
15062 IFSC=PA-1THENPRINT"BIRDIE";A$
15064 IFSC=PATHENPRINT"PAR";A$
15066 IFSC=PA+1THENPRINT"BOGEY";A$
15068 IFSC=PA+2THENPRINT"DOUBLE BOGEY";A
$
15069 IFSC=1THENPRINT"HOLE IN ONE!!!":GO
TO15072
15070 PRINT
15072 PRINT"YOUR TOTAL SO FAR IS"TT
15074 IFQ=0THENPRINT"YOU ARE ON PAR FOR
THE COURSE"
15076 IFQ>0THENPRINT"YOU ARE "Q" OVER PA
R FOR THE    COURSE
15078 IFQ<0THENQ=ABS(Q):PRINT"YOUR TOTAL
 IS"Q"UNDER PAR"
15080 PRINT:PRINT
16008 PRINT"    PRESS THE SPACE"
16010 K$=INKEY$
16020 I$=INKEY$:KD=RND(DD)
16030 DD=DD+1:IFDD>100THENDD=1
16040 IFI$=""THEN16020
16050 IFI$<>" "THEN16020
16060 CLS
16100 IFPL=1THENHO=HO+1:GOTO350
16200 IFPL=2ANDLP=1THENLP=0:HO=HO+1:GOTO
350
16210 IFPL=2ANDLP=0THEN:LP=1:GOTO510
20000 COLOR4
20001 MODE(1):GF=0
20002 IFPA=4ORPA=5THEN20112
20005 FORI=0TO127STEP2
20010 SET(I,8):SET(RND(126),RND(7))
20020 NEXT
20030 FORI=0TO40STEP2
20040 SET(I,31):SET(RND(40),31+RND(31))
20050 NEXT
20060 FORI=86TO127STEP2
20070 SET(I,31):SET(RND(40)+86,31+RND(31
))
20080 NEXT
20090 FORI=31TO63STEP2
20100 SET(40,I):SET(86,I)
20110 NEXT
20111 GOTO20200
20112 IFP2=2THEN20140
20115 FORI=0TO127STEP2
20119 SET(I,8):SET(RND(126),RND(7))
20120 NEXT
20122 FORI=16TO127STEP2
20124 SET(I,31):SET(RND(110)+16,31+RND(3
1))
20126 NEXT
20128 FORI=31TO63STEP2
20130 SET(16,I)
20132 NEXT
20134 GOTO20200
20140 FORI=0TO127STEP2
20142 SET(I,8):SET(RND(126),RND(7))
20144 NEXT
20150 FORI=0TO111STEP2
20152 SET(I,31):SET(RND(110),RND(31)+31)
20154 NEXT
20156 FORI=31TO63STEP2
20158 SET(111,I)
20160 NEXT
20162 GOTO20200
20200 FORI=A-GTOA+G
20210 FORJ=BB-GTOBB+G
20220 SET(I,J)
20225 NEXT:NEXT
20226 COLOR2
20228 FORI=BB-11TOBB:RESET(A,I):NEXT
20232 FORI=BB-11TOBB:SET(A,I):NEXT
20233 FORJ=BB-11TOBB-8
20234 FORI=ATOA+4
20235 SET(I,J):NEXT:NEXT
20236 IFZB=1THEN20265
20238 COLOR2
20240 FORI=C-BTOC+BSTEP2
20250 FORJ=D-BTOD+BSTEP2
20260 SET(I,J)
20264 NEXT:NEXT
20265 IFZJ=1THEN20273
20266 COLOR2
20267 FORI=J1-J3TOJ1+J3STEP2
20268 FORJ=J2-J3TOJ2+J3STEP2
20269 SET(I,J)
20270 NEXT:NEXT
20273 IFZW=1THEN20349
20275 COLOR3
20280 FORI=E-WTOE+WSTEP2
20290 FORJ=F-WTOF+WSTEP2
20300 SET(I,J)
20310 NEXT:NEXT
20349 COLOR4
20350 FORI=SX-2TOSX+2
20360 SET(I,60)
20365 NEXT
20370 FORI=60TO63
20380 SET(SX,I)
20385 NEXT
20970 RETURN
20980 K$=INKEY$
20982 I$=INKEY$:IFI$=""THEN20982
20984 IFI$<>" "THEN20982
20990 RETURN
60000 CLS
60010 PRINT"YOU WERE IN THE WATER AND HA
VE"
60020 PRINT"BEEN REPOSITIONED FURTHER BA
CK"
60030 PRINT"WITH A PENALTY OF 1"
60040 FORI=1TO3000:NEXT
60050 GOTO715
60060 MODE(1)
60070 GS=INT(42/(2*G))
60080 HH=2*(H-A)*GS+63
60090 KK=(K-BB)*GS+31
60093 COLOR4
60095 FORI=12TO106STEP2
60100 SET(I,8):SET(I,55)
60110 NEXT
60120 FORI=8TO55STEP2
60130 SET(12,I):SET(106,I)
60140 NEXT
60145 COLOR2
60150 FORI=12TO31
60160 SET(63,I)
60165 NEXT
60170 FORI=63TO75
60180 FORJ=12TO18
60190 SET(I,J)
60200 NEXT:NEXT
60210 FORI=63-GSTO63+GS
60220 FORJ=31-GS/2TO31+GS/2
60230 SET(I,J)
60240 NEXT:NEXT
60243 COLOR4
60245 K$=INKEY$
60246 I$=INKEY$
60250 SET(HH,KK):SET(HH+1,KK)
60270 IFI$=""THEN60246
60280 IFI$<>" "THEN60246
60285 IFDI<=.5THEN15000
60290 GOTO8004
60300 PRINT@176,"90"
60310 PRINT@208,"."
60312 PRINT@240,"."
60314 PRINT@272,"."
60320 PRINT@297,"180..BALL....0"
60330 PRINT@336,"."
60332 PRINT@368,"."
60334 PRINT@400,"."
60340 PRINT@432,"270"
60360 RETURN
```

KNIGHTS CROSS

The program is purely graphics and works as follows:

Line 16 sets random colour.

Lines 30-60 creates what I call an inverted German Cross in multi colours.

Lines 90-200 draw a circle in the cross.

Lines 345-370 draw a square.

Line 370 pauses to display the image.

The end result looks like the 'Knights Cross with oak leaves' just like the Germans issued their war heroes.

It shows how we can use the capabilities of the VZ200 todraw very intricate designs by allowing the composition and placement of the A Z Y in the lines 40-43 and 100-170, i.e. A+60 change to A-60 or A+60, 30+Y change to 30+Y,A-60 all sorts of wonderful patterns can be created.

G. Lucas
Boroko PNG

```
1 REM"KNIGHT'S CROSS"
2 REM"LUKE LUKAS,PAPUA NEW GUINEA"
3 REM"OCTOBER,1983"
10 CLS
14 MODE(1)
15 FORR=1TO24
16 C=RND(3)+1:COLORC
20 FORA=-RTOR
30 Y=SQR(A*A+R*R):Y= INT(Y-.5)
40 SET(A+60,30+Y)
41 SET(60-A,34-Y)
42 SET(Y+65,32-A)
43 SET(55-Y,32+A)
60 NEXT:NEXT
70 FORR=1TO12
80 FORA=-RTOR
90 Y=SQR(R*R-A*A):Y=INT(Y-.5)
95 C=RND(3)+1:COLORC
96 SET(A+60,30+Y):SET(A+60,30-Y)
100 SET(A+12,30+Y)
110 SET(A+12,30-Y)
120 SET(A+114,30+Y)
130 SET(A+114,30-Y)
140 SET(A+60,13+Y)
170 SET(A+60,50+Y)
200 NEXT:NEXT
300 C=RND(3)+1:COLORC
345 FORX=0TO127:FORY=0TO1:
SET(X,Y):NEXT:NEXT
350 FORX=0TO127:FORY=62TO63:
SET(X,Y):NEXT:NEXT
360 FORX=0TO1:FORY=0TO63:
SET(X,Y):NEXT:NEXT
370 FORX=126TO127:FORY=0TO63:
SET(X,Y):NEXT:NEXT
400 FORT=1TO2000:NEXT
410 COPY
1000 GOTO14
```

PROGRAMS FOR SINCLAIR

GRAND PRIX

(For ZX81, 1K)

Grand Prix involves driving your racing car (an inverse 'H') down a convincing simulation of a racing course, while dodging other cars and people who are cleverly disguised. The object of the game is to last as long as possible.

The keys '5' and '8' move you left and right. There is a high score feature and, after a game, press any key to play again.

This program uses a number of unusual techniques. First of all, it uses a bit of machine code POKEd onto the first REM statement to speed things up. When called (by USR 16514), it returns a code relating to the character in the correct PRINT position. This replaces the line: PEEK (PEEK 16398 + 256* PEEK 16399)

Also, in my article, 'Larger Screen' in Your Computer July 1983 issue, POKE 16418,0 was used to obtain a 24 line screen. This location can also be used to make the screen smaller, and so save memory.

POKE 16418,5 would change the screen to 32 by 18 lines – PRINT AT 19,0;"Jason" would result in an out of screen error report. The SCROLL command is also changed to work from this line.

The scoring is also unusual, since the line NEXT S is used instead of the line LET S = S + 1 which uses up more memory and is slower.

In the listing, △ is a space and 'gr.A' means the graphic character on that key.

My best score is 413. Can you beat that?

Jason Teh
Doncaster VIC

```
 1 REM 0000000
 POKE 16514,42
 POKE 16515,14
 POKE 16516,64
 POKE 16517,78
 POKE 16518,6
 POKE 16519,0
 POKE 16520,201
 2 POKE VAL "16418", VAL "5"
10 LET H=NOT PI
20 LET A=VAL "8"
30 LET B=VAL "5"
40 CLS
50 LET D=INT PI
60 FOR S=0 TO 10
70 SCROLL
80 PRINT TAB D; "inverse space, gr. A, gr. A,
gr. A, inverse space"
90 IF RND>.7 THEN PRINT AT 18,D+RND*2+1; "gr. S"
100 NEXT S
110 LET B=B-(INKEY$="5")+(INKEY$="8")
120 PRINT AT A,B;
130 IF USR 16514<>CODE "gr. A" THEN GOTO 180
140 PRINT "inverse H"
150 LET X=RND
160 LET D=D+(X>.5 AND D<10)-(X<.5 AND D>0)
170 GOTO 70
180 PRINT "H"
190 IF H<S THEN LET H=S
200 PRINT "S=";S;"△△H=";H
210 PAUSE VAL "4E4"
220 GOTO VAL "20"
```

ENTER DIRECTLY ON THE KEYBOARD.

ASTEROID DODGE

(for ZX81, 1K)

Asteroid Dodge is an addictive game for the 1K ZX81. The aim is to safely pilot your ship through space dodging the asteroids hurtling towards you.

Your ship is always moving left and cannot stay still. Pressing any key will move you right. Also, if you move too far to the side of the screen, your ship will disappear and appear on the other side, which makes the game harder.

There is a high score and scoring mechanism. At the end of a game, just press any key to restart. Also, the game uses a short machine code routine POKEd onto a REM statement to speed things up. When called, the code value of the character in the current PRINT position is returned.

Have fun!!!

Jason Teh
Doncaster VIC

```
 1 REM 0000000
POKE 16514,42
POKE 16515,14
POKE 16516,64
POKE 16517,78
POKE 16518,6
POKE 16519,0
POKE 16520,201
10 LET H=0
20 LET S=0
30 CLS
40 LET A=10
50 LET B=A
60 PRINT AT 21,RND *18;"*"
70 SCROLL
80 LET B=B-1
90 IF INKEY$<>"" THEN LET B=B+2
100 LET B=B+(-B AND B>18)+(20 AND B<0)
110 PRINT AT A,B;
120 IF USR 16514=CODE "*" THEN GOTO 160
130 PRINT "V"
140 LET S=S+1
150 GOTO 60
160 IF H<S THEN LET H=S
170 PRINT "inverse . =";S;"△△inverse H=";H
180 PAUSE 4E4
190 GOTO 20
```

N.B. To make the game harder, change line 60 to :

```
60 PRINT AT 21,RND*18; "**"
```

ESCAPE
(For ZX81, 16K)

Here's a short program – a game which starts easy and as you progress becomes more difficult.

The object of the game is to dodge the black squares and to travel through the time gate which appears at random. If you do make it, more black squares appear randomly. Your score goes by steps of your input and rounds go up by one every time you reach the other side. Skill is obtained by score, rounds and level. To move your star use the keys A for up and Z for down.

Garry Wilson
Higgins ACT

```
00010 LET B$="(Graphics space 32 times)"
00020 LET S=0
00030 LET D=10
00040 LET U=1
00050 LET R=0
00060 GOTO 170
00070 PRINT AT D,U;" "
00080 LET U=U+1
00090 LET S=S+L
00100 LET K$=INKEY$
00110 LET D=D+(K$="Z")-(K$="A")
00120 PRINT AT D,U;
00130 IF PEEK(PEEK 16398+PEEK 16399*256)=128 THEN GOTO 390
00140 PRINT AT D,U;"*"
00150 IF U>30 THEN GOTO 240
00160 GOTO 70
00170 PRINT AT 5,8;" * E S C A P E * "
00180 PRINT AT 10,8;"Input level(1-5)"
00190 INPUT L
00200 IF L<>INT (L) OR L<=0 OR L> 5 THEN GOTO 190
00210 CLS
00220 PRINT B$
00230 PRINT AT 21,0;B$
00240 LET X=2
00250 LET R=R+1
00260 LET X=X+1
00270 LET A1=RND*(20
00280 LET A2=RND*(24)+2
00290 PRINT AT A1,A2;"(Graphics space 1 time)"
00300 IF X=L THEN GOTO 330
00310 GOTO 230
00320 FOR E=0 TO 20
00330 PRINT AT E,31;"(Graphics space 1 time)"
00340 NEXT E
00350 LET E2=RND*20+1
00360 PRINT AT E2,31;" "
00370 LET U=1
00380 GOTO 70
00390 CLS
00400 PRINT AT 5,5;"GAME OVER"
00410 PRINT AT 10,5;"SCORE ";S
00420 PRINT AT 15,5;"ROUND ";R
00430 PRINT AT 20,5;"SKILL ";L*R+S
00440 PRINT"PRESS 'N' KEY FOR ANOTHER GAME"
00450 IF INKEY$<>"N"THEN GOTO 450
00460 CLS
00470 RUN
```

N.B. To make the game harder, change line 60 to :

```
60 PRINT AT 21,RND*18; "**"
```

ZX81 SKETCH
(For ZX81, 1K)

There are many 'drawing' programs for the ZX81, but many are wasteful of memory. While this is alright for a 16K ZX81 1K owners are much disadvantaged. I have sought to remedy this, and have produced "Sketch" for the 1K ZX81, using the PLOT command to get a resolution of 40 by 40.

There are 8 directions in which you can move and you can rubout as well as draw. The key directions are with the listing.

To get into DRAW mode, press '9' and to RUBOUT, press '0'. If you own a 16K ZX81, change and add the extra lines. You now have some bonus commands as well as a resolution of 64 by 44. Pressing 'Z' will dump the screen to the printer, and will let you continue drawing. "C" will now clear the screen. Finally, pressing 'S' will SAVE the screen and the program onto tape, and will let you continue drawing the same picture at a later date. Before pressing 'S', make sure the tape recorder is recording!!

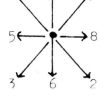
Jason Teh
Doncaster VIC

```
 10 LET Z=1
 20 LET X=0
 30 LET Y=X
 40 LET A$=INKEY$
 50 IF A$="0" THEN LET Z= -Z
 60 LET X=X+(A$="8" OR A$="1" OR A$="2")-
    (A$="5" OR A$="3" OR A$="4")
 70 LET Y=Y+(A$="7" OR A$="1" OR A$="4")-
    (A$="6" OR A$="2" OR A$="3")
 80 LET X=X+(X<0)-(X>40)
 90 LET Y=Y+(Y<0)-(Y>40)
100 UNPLOT X,Y
110 PLOT X,Y
120 IF Z=1 THEN UNPLOT X,Y
130 GOTO 40
```

If you own a 16k ZX81, change and add the following:

```
 80 LET X=X+(X<0)-(X>63)
 90 LET Y=Y+(Y<0)-(Y>43)
130 IF A$="Z" THEN COPY
140 IF A$="C" THEN CLS
150 IF A$="S" THEN SAVE "SKETCH"
160 GOTO 40
```

KEY DIRECTIONS --



ATTR FILL

(For Spectrum)

This program is for a ZX Spectrum with any memory size. One annoying fact that I find when using the Spectrum is that if you want the PAPER, INK, FLASH or BRIGHT commands to work globally (the whole screen), you have to clear the screen first with CLS and so destroy the display, which might have taken a long time to set up. To overcome this, I have written a short machine code routine, only 18 bytes long, which changes the screen instantly without clearing the screen.

To use it,just enter the appropriate colour commands such as: PAPER 6: INK 1: FLASH 1 and call the routine. The whole screen should then become yellow, with blue lettering and everything flashing. The screen displays will still be there.

You can also do this by POKING the attribute number into address 23693, which in this case would be 177, and then calling the routine.

There are two parts to the listing. The first POKEs the machine code stored in a DATA statement, above RAMTOP and the second is a demonstration program.

As listed, the program is for a 16K Spectrum but to change it for a 48K Spectrum, change every address 32582 to 65349 and every 32583 to 65350. You call the routine by LET L = USR 32583 – 16K or LET L = USR 65350 – 48K.

First type in Listing 1 and RUN it. Save the code by SAVE "ATTR FILL"CODE 32583,18 and VERIFY it. NEW the machine and type in the demonstration program. Remember to CLEAR 32582 before LOADING.

Jason Teh
Doncaster Vic

```
10 CLEAR 32582: LET X=32582
20 FOR N=X+1 TO X+18
30 READ A: POKE N,A
40 NEXT N
50 DATA 33,0,88,1,192,90,58,141,92,119,
   35,167,237,66,9,32,248,201
```

Demonstration.

```
10 FOR N=0 TO 255 STEP 2
20 PLOT 0,0: DRAW N,RND*175
30 NEXT N
40 FOR N=0 TO 7
50 PAPER N: INK 7-N
60 LET L=USR 32583
70 PAUSE 10
80 NEXT N
90 GOTO 40
```

CHEMISTRY

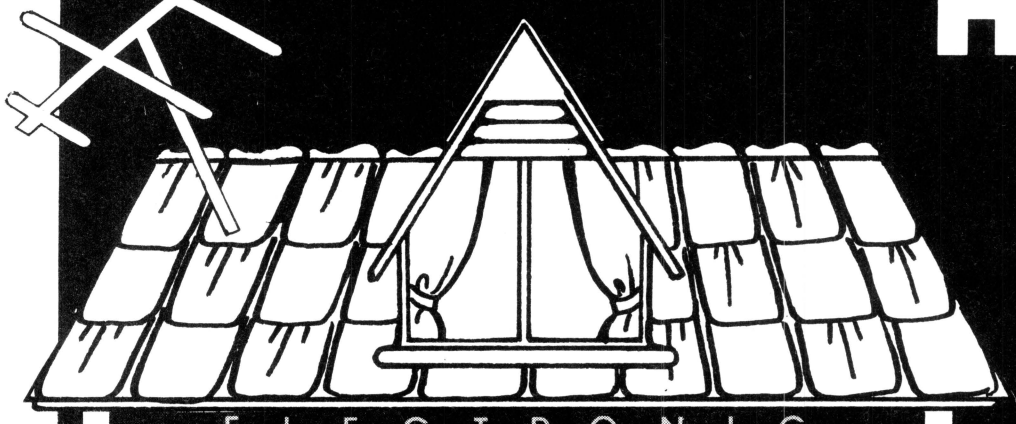
(For ZX81, 1K)

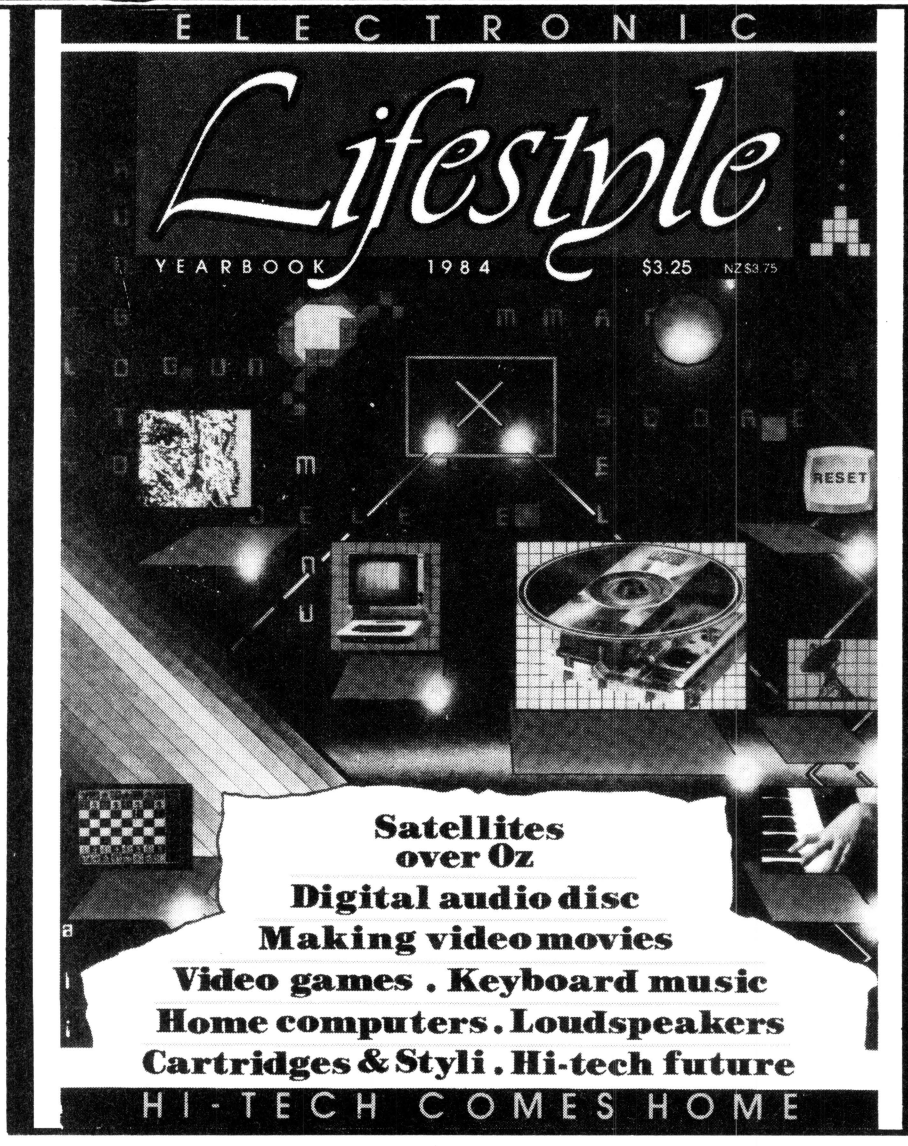
This program tests your knowledge of the first ten chemical elements. Run the program and the computer will randomly print the name of one of the elements and asks you to input the atomic number and then the symbol for that element. The computer will tell you whether your answers are correct or will give you the correct answer.

D.W. Moore
North Geelong VIC

```
 2 PRINT "CHEMISTRY" inverse
 5 PAUSE 150
 6 CLS
10 GOSUB INT (RND * 10) * 2 + 100
15 PRINT A$ (5 TO)
16 FOR F=1 TO 3 STEP 2
17 IF F=1 THEN PRINT "ATOMIC NUMBER?  ";
18 LET C$= "  " two spaces
19 IF F=3 THEN PRINT "SYMBOL?  ";
20 INPUT B$
27 LET C$ (1) = B$ (1)
28 IF LEN B$>= 2 THEN LET C$ (2) = B$ (2)
40 IF C$ <> A$ (F TO F+1) THEN GOTO 80
50 PRINT "YES  ";C$
60 GOTO 85
80 PRINT "NOT ";C$;" ...."; A$ (F TO F+1)
85 NEXT F
90 GOTO 5
100 LET A$ = "1 H HYDROGEN"
101 RETURN
102 LET A$ = "2 HEHELIUM"
103 RETURN
104 LET A$ = "3 LILITHIUM"
105 RETURN
106 LET A$ = "4 BEBERYLLIUM"
107 RETURN
108 LET A$ = "5 B BORON"
109 RETURN
110 LET A$ = "6 C CARBON"
111 RETURN
112 LET A$ = "7 N NITROGEN"
113 RETURN
114 LET A$ = "8 O OXYGEN"
115 RETURN
116 LET A$ = "9 F FLUORINE"
117 RETURN
118 LET A$ = "10NENEON"
119 RETURN
```

HI-TECH COMES HOME!





All of a sudden, the future is NOW! Here's a magazine to help you understand — and enjoy — the latest technological advances.

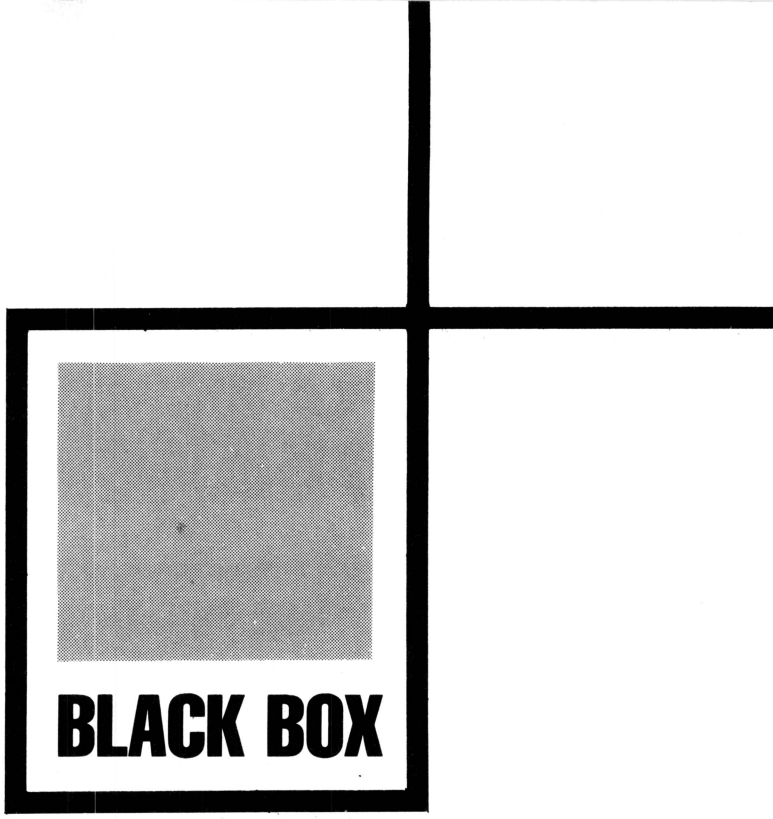
All around us, breakthroughs in technology are occuring. Changes are being made to many facets of our lives. Electronics Lifestyle is all about your lifestyle. It reports on advances in computing . . . entertainment . . . communications. Learn how to make the most of the new breakthroughs — and how to make them work for you!

$3.25

at your news stand

PROGRAMS FOR SIGMA OKI

SIGMA OKI



My program is a variation of the game 'Black Box' invented by Dr. Eric Solomon. It is a game of logic that can easily be converted to run on a Peach because the two machines are virtually identical.

The object of black box is to find four protons in a box. The paths of electrons shot into the box give the clues. Because the red light emission of the electron is only temporary it becomes necessary to have a working space to put down your ideas. This is provided in the form of a second board, disconnected from play.

The number of shots you have used to find the correct

```
1 CLS:COLOR 4,0,0,7:WIDTH 40,25
2 DEF CHR$(123)="080810E010080808":DEF CHR$(148)="80C0E0F0F8FCFEFF":DEF CHR$(149
)="0103070F1F3F7FFF":REM  ◥  ◣  ◢       Respectavly.
3 PRINT"Do you want instructions?"
4 P=PEEK(1424):IF P=0 THEN P=RND:GOTO 4
5 IF P=89 OR P=121 THEN 45
6 IF PEEK(1424)=0 THEN X=RND:GOTO 6
7 CLS
8 X$=INKEY$:FOR X=329 TO 570 STEP 30
9 LINE (X,47)-(X,151),PSET,1
10 NEXT
11 FOR Y=47 TO 152 STEP 13
12 LINE (329,Y)-(569,Y),PSET,1
13 NEXT
14 COLOR 6:PRINT "     'CLS'                'HOME'":COLOR 4
15 LOCATE 0,23:PRINT "PRESS THE SPACE BAR TO FIRE AND":PRINT "THE RETURN KEY TO
SHOW THE PROTONS";
16 FOR I=1 TO 8:LOCATE 3,I+7:PRINT "●●●●●●●●":NEXT
17 DIM A(9,9):FOR X=0 TO 9:A(X,0)=2:A(X,9)=2:A(0,X)=2:A(9,X)=2:NEXT
18 FOR I=1 TO 4:N(I)=INT(RND*8)+1:M(I)=INT(RND*8)+1:NEXT
19 FOR I=1 TO 3:FOR O=I+1 TO 4:IF N(O)=N(I) AND M(O)=M(I) THEN I=20:O=20
20 NEXT:NEXT:IF I>10 THEN 18
21 FOR X=1 TO 4:A(N(X),M(X))=1:NEXT:W=2
22 GOSUB 137
23 CC=0:Q1=0:W1=W:IF W=0 THEN XX=0:YY=1:GOTO 27
24 IF W=9 THEN XX=0:YY=-1:GOTO 27
25 IF Q=0 THEN XX=1:YY=0:GOTO 27
26 XX=-1:YY=0
27 CC=CC+1:Q1=Q1+XX:W1=W1+YY:C=A(W1,Q1):IF CC>59 THEN BEEP:BEEP:BEEP:GOTO 44
28 IF C=2 THEN 39
29 IF C=1 THEN 43
30 IF YY=0 THEN C1=A(W1-1,Q1):C2=A(W1+1,Q1):GOTO 35 ELSE C1=A(W1,Q1-1):C2=A(W1,Q
1+1)
31 IF C1=1 AND C2=1 THEN 27
32 IF C1=1 THEN XX=1:YY=0:GOTO 27
33 IF C2=1 THEN XX=-1:YY=0
34 GOTO 27
35 IF C1=1 AND C2=1 THEN 27
36 IF C1=1 THEN YY=1:XX=0:GOTO 27
37 IF C2=1 THEN YY=-1:XX=0
38 GOTO 27
39 CL=2
40 IF Q1=0 THEN K=W1*13+41:LINE (329,K)-(200,K),PSET,CL:GOTO 43 ELSE IF Q1=9 THE
N K=W1*13+41:LINE (569,K)-(639,K),PSET,CL:GOTO 43
41 IF W1=0 THEN K=Q1*30+314:LINE (K,47)-(K,0),PSET,CL:GOTO 43
42 IF W1=9 THEN K=Q1*30+314:LINE (K,151)-(K,199),PSET,CL
43 IF CL=2 THEN CL=0:PAUSE 3:GOTO 40
44 CN=CN+1:BEEP:LOCATE 0,0:PRINT CN;:GOTO 22
45 CLS:COLOR 7
46 PRINT
47 PRINT
48 PRINT "
49 PRINT "
50 PRINT "
51 PRINT "
52 PRINT "
53 PRINT "
54 PRINT "
55 PRINT
56 PRINT
57 PRINT
58 PRINT
59 PRINT "
60 PRINT "
61 PRINT "
62 PRINT "
63 PRINT "
64 PRINT "
65 PRINT "
66 PRINT
67 FOR P=1 TO 800:COLOR ,,,1:COLOR ,,,2:COLOR ,,,4:NEXT:
   COLOR 4,,,7
68 CLS:PRINT "This game is based on the principles of"
69 PRINT "a physics experiment. Basicly there are"
70 PRINT "four protons in a square box, 8 by 8 in"
71 PRINT "area. You are the experimenter and need"
72 PRINT "to know where they are.":PRINT
73 PRINT "To do this you fire a charged particle,"
74 PRINT "into the side of the box and note where"
75 PRINT "it emerges. Move the electron around   "
76 PRINT "the border and when it is in position,"
77 PRINT "press the space bar. It will then move"
78 PRINT "unseen guided by the four protons."
79 PRINT
80 PRINT "The protons will absorb the electron if"
81 PRINT "it collides head on. When it passes the"
82 PRINT "proton in the adjacent co-ordinate, it "
83 PRINT "will change direction by 90 degrees. If"
```

SIGMA OKI

position of all the protons is the measure of how good you are. Seven shots is reasonable but teens are common. It is possible to change the number of protons by changing lines 18, 19 and 21. The program is written in Microsoft Basic so it can be adapted to most machines. It's well worth the effort. Further instructions on play are in the program but these are not necessary for it to run. Have fun and don't strain anything!

Tony Hinde
Tarragindi Qld

```
84 PRINT "it Passes between two Protons then its"
85 PRINT "Path is uneffected."
86 PRINT
87 PRINT "When the electron emerges from the box"
88 PRINT "it leaves a beam of red light for a"
89 PRINT "few seconds."
90 COLOR 5:PRINT "HIT ANY KEY TO CONTINUE";:COLOR 4
91 IF PEEK(1424)=0 THEN X=RND:GOTO 91 ELSE X$=INKEY$
92 CLS
93 PRINT "Here are some example shots....."
94 PRINT " ●●●●●●X●                          ●●●●●●●●"
95 PRINT " ●X↗→→↘●    X -Proton              ●●●●●●●●"
96 PRINT " ●●↑●●●↓●                          ●●●●X●●●"
97 PRINT " ●●↑●●X↳——→ -Exit          ←——←←←←↖X●●"
98 PRINT " ●●↑●●●●●                          ●●●●↑●●●"
99 PRINT " ●●↑●●●●●                          ●●●●↑●●●"
100 PRINT " ●●↖←←←←←←    ←Entry Point        ●X↗↙●●●"
101 PRINT " ●●X●●●●●                          ●●↑●X●●●"
102 PRINT "                                     ↑     "
103 COLOR 5:PRINT "HIT ANY KEY"
104 PRINT "TO CONTINUE";:COLOR 4:PRINT "                    ↑"
105 PRINT "                                        |"
106 PRINT "                                        |"
107 PRINT " ●●●●●●●●       ●●●●●●X●       ●●●●●●↑●"
108 PRINT " ●●XX●●●●       ●X↗→→↘●       ●X↗→→↘●"
109 PRINT " ●●●↑●●●●       ●●↑●●●↓●       ●●↑●●●X●"
110 PRINT " ●●●↑●●●●       ●●↑●●●↓●       ●●↑●●●●●"
111 PRINT " ●●X↑X●●●       ●●↑●●●↓●       ●●↑●●●●●"
112 PRINT " ●●●↑●●●●       ●●↖←←←←X       ●●↖←←←↖X"
113 PRINT " ●●●↑●●●●       ●●X●●●↑●       ●●X●●●↑●"
114 PRINT " ●●●↑●●●●       ●●●●●●↑●       ●●●●●●↑●"
115 PRINT "    ↑              ↑              ↑"
116 IF PEEK(1424)=0 THEN X=RND:GOTO 116 ELSE X$=INKEY$
117 CLS:PRINT "To assist you in your experiment a board";
118 PRINT "has been Provided for You to use as a"
119 PRINT "guide. You can set up Protons on it so"
120 PRINT "You can best judge where to shoot next."
121 PRINT "To use the bourd You Press the CLS key,"
122 PRINT "then You can move about the board by the";
123 PRINT "arrow keys. To set a Proton Press the"
124 PRINT "INS key. to reset it Press the DEL key."
125 PRINT "You may Press any key and that simbol or";
126 PRINT "letter will appear on the board. This"
127 PRINT "way You may cancel Places that You know"
128 PRINT "are not Protons. You can also rank the"
129 PRINT "Probability of a Proton being in a spot.";
130 PRINT "When you want to get back to the exPer-"
131 PRINT "iment You Press the HOME key."
132 PRINT:PRINT:COLOR 5:PRINT"PRESS ANY KEY TO CONTINUE":COLOR 4
133 GOTO 6
134 FOR X=1 TO 8:FOR Y=1 TO 8:IF A(X,Y)=1 THEN PAINT ((Y)*30+315,(X)*13+40),2,1
135 NEXT:NEXT
136 IF PEEK(1424)=0 THEN 136 ELSE RUN
137 X2=315+Q*30:Y2=41+W*13:X1=X2:Y1=Y2
138 P=PEEK(1424):IF (P<28 OR P>32) AND P<>13 AND P<>12 THEN U=7-U:PSET(X1,Y1,U):
GOTO 138
139 IF P=12 THEN GOSUB 151:GOTO 138
140 IF P=13 THEN 134
141 BX=Q:BY=W:IF P=28 THEN Q=Q+1
142 IF P=29 THEN Q=Q-1 ELSE IF P=30 THEN W=W-1 ELSE IF P=31 THEN W=W+1
143 IF P=32 THEN IF (Q=0 OR Q=9) AND (W=0 OR W=9) THEN 147 ELSE RETURN
144 IF Q<0 OR Q>9 OR W>9 OR W<0 THEN 147
145 IF Q>0 AND Q<9 AND W>0 AND W<9 THEN 147
146 GOTO 148
147 Q=BX:W=BY:GOTO 138
148 X1=315+Q*30:Y1=41+W*13
149 PSET (X1,Y1,7):PSET (X2,Y2,0):X2=X1:Y2=Y1
150 GOTO 138
151 J=0:K=0:GOTO 167
152 P=PEEK(1424):IF P=0 THEN LOCATE J+3,K+8:PRINT" "CHR$(29)K$;:GOTO 152
153 IF P=11 THEN RETURN
154 IF P=8 THEN LOCATE J+3,K+8:PRINT "●";:GOTO 167
155 IF P=18 THEN LOCATE J+3,K+8:PRINT"X";:GOTO 167
156 IF P>27 AND P<32 THEN 159
157 IF P>31 THEN LOCATE J+3,K+8:PRINT CHR$(P);:GOTO 167
158 GOTO 152
159 IF P=28 THEN J=J+1:GOTO 163
160 IF P=29 THEN J=J-1:GOTO 163
161 IF P=30 THEN K=K-1:GOTO 163
162 K=K+1
163 IF K>7 THEN K=7
164 IF K<0 THEN K=0
165 IF J<0 THEN J=0
166 IF J>7 THEN J=7
167 K$=CHR$(SCREEN(J+3,K+8)):GOTO 152
```

WORD COUNT FOR HITACHI PEACH

Phillip Cookson
Doveton NSW

After reading Les Bell's "Understanding Assembler" tutorial in the October 1983 issue of "Your Computer", I was inspired to write a BASIC program to count the words in a word processing data file (Les suggested to write a word counting program in a higher level language before attempting a machine code version). The program is written in Microsoft BASIC. Although the program is slow, it works correctly and will be of use to anyone who uses the HiWriter word processing software. The program performs a word count of a text file created by HiWriter (which currently has no facility for performing a word count). It ignores embedded format commands and remarks giving an accurate word count.

```
100 ' WORD COUNT      By  Philip Cookson      November, 1983
110 '
120 ' This program performs a word count on a data file created by the
130 ' HiWriter word processing package.  Words are defined to be groups
140 ' of letters A-Z or a-z separated by spaces or by a carriage return.
150 ' This program ignores control characters, or words appearing on an
160 ' @ command line.
170 '
180 ON ERROR GO TO 830
190 CLS:WIDTH 80:NWORD=0:DRVE$="1:"
200 LINE(0,0)-(639,199),PSET,1,B:COLOR 5
210 LOCATE 31,2:PRINT "W O R D  C O U N T"
220 LOCATE 31,3:PRINT "------------------"
230 LOCATE 25,23:PRINT "### Default Drive is  1: ###"
240 '
250 ' Input name of data file that is to be counted
260 '
270 COLOR 6:LOCATE 8,5
280 PRINT "Please input the name of the data file that you wish to count."
290 COLOR 7:LOCATE 35,7:INPUT"",FLNME$
300 '
310 ' Place the input in a valid format
320 '
330 IF RIGHT$(FLNME$,4)<>".DAT" THEN FLNME$=FLNME$+".DAT"
340 IF LEFT$(FLNME$,2)<>"1:" OR LEFT$(FLNME$,2)<>"0:" THEN FLNME$=DRVE$+FLNME$
350 '
360 ' Print name of data file being searched
370 '
380 LOCATE 6,5:PRINT SPC(70)
390 COLOR 6:LOCATE 22,5:PRINT "Data file currently being searched"
400 COLOR 7:LOCATE 33,7:PRINT FLNME$
410 '
420 ' Open data file to be searched
430 '
440 OPEN "I",#1,FLNME$
450 COLOR 3
460 LOCATE 30,10:PRINT " Number of words "
470 LOCATE 30,14:PRINT "Count in progress"
480 '
490 ' Search for letters A-Z or a-z
500 '
510 ACHAR$=INPUT$(1,#1)
520 IF ASC(ACHAR$)>64 AND ASC(ACHAR$)<91 THEN GO TO 580
530 IF ASC(ACHAR$)>96 AND ASC(ACHAR$)<123 THEN GO TO 580
540 IF ACHAR$="@" THEN GO TO 650 ELSE GO TO 510
550 '
560 ' Subroutine to count words, and locate the end of the word
570 '
580 NWORD=NWORD+1:LOCATE 37,12:PRINT NWORD
590 ACHAR$=INPUT$(1,#1)
600 IF ACHAR$=" " THEN GO TO 510
610 IF ACHAR$=CHR$(13) THEN GO TO 510 ELSE GO TO 590
620 '
630 ' Subroutine to ignore all words appearing on an @ command line
640 '
650 ACHAR$=INPUT$(1,#1)
660 IF ACHAR$="@" THEN GO TO 510
670 IF ACHAR$=CHR$(13) THEN GO TO 510 ELSE GO TO 650
680 '
690 ' Print the word count on the screen
700 '
710 LOCATE 37,12:PRINT NWORD
720 COLOR 2:LOCATE 30,14:PRINT "Counting Completed":BEEP
730 CLOSE #1
740 '
750 ' Search another data file ?
760 '
770 COLOR 4:LOCATE 15,20
780 PRINT "Do you wish to search another data file  (Y/N) ? "
790 ANS$=INKEY$:IF ANS$="" THEN GO TO 790
800 IF ANS$="n" OR ANS$="N" THEN COLOR 7:CLS:END
810 IF ANS$="y" OR ANS$="Y" THEN GO TO 190 ELSE BEEP:GO TO 790
820 '
830 ' Error handling subroutine
840 '
850 ' End of data file detection
860 IF ERR=54 THEN RESUME 710
870 '
880 ' File not found error
890 IF ERR<>63 THEN GO TO 950
900 COLOR 2:LOCATE 32,9
910 BEEP:INPUT WAIT 920;5,"FILE NOT FOUND";DUM$
920 RESUME 190
930 '
940 ' Device Unavailable error
950 IF ERR<>60 THEN GO TO 1010
960 COLOR 2:LOCATE 32,9
970 BEEP:INPUT WAIT 980;5,"DEVICE UNAVAILABLE";DUM$
980 RESUME 190
990 '
1000 ' Miscellaneous error
1010 CLS:LOCATE 5,12
1020 BEEP:PRINT "ERROR CODE",ERR,"ON LINE",ERL:END
```

CAMEL FOR CASIO PB-100

The game 'Camel' is originally from the book 'Basic Computer Games' by David Ahl, although this version was written on a train trip between northern NSW and Sydney.

The aim of the game is to travel 200 miles across a hostile desert while the pygmies are chasing you. In order to fit it into a PC-100 with the RAM pack (it takes 1399 steps) the messages have been abbreviated. For example:

L. 120 – distance pygmies are behind;

L. 130 – distance you have travelled;

L. 610 – turns you can go without a drink;

L. 600 – number of days the camel can travel without a rest.

The instructions have been included so they may be written down if necessary before you play.

The display frequently halts and the EXEC key must be hit to continue (this is faster than letting it print out long messages on its own).

During the course of the game, variables, such as drinks left or camel days left, will vary according to your instructions so a STATUS CHECK (5) will indicate whether it is necessary to stop for the night or take a drink.

Variables used are:

C...distance covered by the player

D...distance covered by the pygmies

Z...turns you can go without a drink

S...number of drinks left

F...'camel days' left

G...skill level

Y...command choice

P & R are used as counters.

Linda McGarry
Kentucky NSW

```
10 PRINT "CAMEL"
20 INPUT "INSTRUCTION",$:IF
    MID(1,1)="Y";GOSUB 1000
30 GOSUB 1030
40 INPUT "LEVEL 1-5",G
50 IF C>199 THEN 960
60 Z=Z-1:IF Z<=1;PRINT "GET DRINK"
70 IF Z<0 THEN 940
80 P=P+1:X=INT((3+G)*RAN#+2.5)
90 IF P<4 THEN 130
100 D=D+X:IF D<C THEN 120
110 PRINT "Captured by pygmies":GOTO
     1100
120 PRINT "Pygmies";C-D;" m"
130 PRINT "You";C;" m"
140 IF S=1;PRINT "Oasis..now"
150 R=0:INPUT "COMMAND",Y:IF Y<1 THEN
     150
160 IF Y>6 THEN 150
170 GOTO Y*100+100
200 S=S-1:Z=4:IF S<0 THEN 940
210 GOTO 140
300 PRINT "Mod. Pace":F=F+1:IF F>7 THEN
     950
310 GOTO 800
320 X=INT(10*RAN#):C=C+X
330 GOTO 50
400 PRINT "Fast Pace":F=F+3:IF F>7 THEN
     950
410 GOTO 800
420 X=INT(RAN#*10)*2:C=C+X
430 GOTO 50
500 PRINT "Good Idea":F=0:IF G>3;S=S-1:
     IF S<0 THEN 940
510 GOTO 60
600 PRINT "Camel";7-F;" days",S-1;"
     drinks"
610 PRINT Z;" turns w/o"
620 S=S-1:Z=4:IF S<0 THEN 940
630 GOTO 140
700 T=INT(RAN#*10):IF T#1 THEN 940
710 PRINT "Found Unconscious":S=3:Z=4:GO
     TO 50
800 A=INT(RAN#*100):IF A>5 THEN 910
820 PRINT "Captured by Berbers","CHOICES
    :"
830 PRINT "1)Attempt escape","2)Wait for
     ransom"
840 INPUT "ACTION",X:IF X=2 THEN 880
850 X=INT(RAN#*10):IF X<5 THEN 870
860 PRINT "ESCAPED!!":GOTO 50
870 PRINT "Mortally Wounded":END
880 X=INT(RAN#*100):IF X<=24;PRINT
     "Ransom paid":GOTO 50
890 PRINT "Wait...":R=R+1:IF R>3;PRINT
     "Cell key lost":END
900 GOTO 840
910 A=INT(RAN#*10):IF A>2 THEN Y*100+120
920 PRINT "Oasis...":Z=4:F=F-2:S=6:IF
     F>7;F=7
930 GOTO Y*100+130
940 PRINT "Terminal dehydration":END
950 PRINT "Killed camel":END
960 PRINT "**YOU WIN**","Pygmies
     pleased":END
1000 PRINT "CHOICES:","1)DRINK","2)MODER
     ATE SPEED"
1010 PRINT "3)FULL SPEED","4)STOP FOR
     NIGHT"
1020 PRINT "5)STATUS CHECK","6)HOPE FOR
     HELP":RETURN
1030 VAC
1040 Z=5:S=6:RETURN
1100 X=INT(RAN#*100):PRINT "CHOICES:"
1110 PRINT "1)Fight","2)Run"
1120 INPUT "ACTION",V:IF V=2 THEN 1150
1130 IF X>80;D=C-20:GOTO 50
1140 PRINT "Kneecaps attacked":END
1150 IF X>60;PRINT "Escaped...":D=C-
     10:GOTO 50
1160 PRINT "Pygmies will feast":END
```

Ohio HIGH-SPEED TRIGONOMETRIC FUNCTIONS

This is a program for Forth on the Ohio Scientific Computers, but the programs are designed to operate on any Forth machine. The program provides high speed trig functions for real time applications.

It is possible to obtain trig functions from the floating point system used by BASIC in ROM but these routines are slow when needed for real time plotting of graphical information. These programs provide a source of values for Sine x to plot circles in real time.

The routine is called using: 'n SIN' where n is a signed integer number on the top of the stack. It returns a signed integer number to the stack. This value is SIN n degrees times 10,000. Values for Cosine are obtained in a similar manner, that is, 'n COS'.

John Lindsay
Trinity Gardens SA

```
SCR # 15
  0 ( TRIG FUNCTIONS ---- J.S.LINDSAY 27/2/83 )
  1 FORTH DEFINITIONS DECIMAL
  2 91 ()DIM SINTABLE ( SET UP 91 ELEMENT ARRAY INC. 0 )
  3 ( START OF LOOK UP TABLE )
  4 0000 0175 0349 0523 0698 0872 1045 1219 1392 1564
  5 1736 1908 2079 2250 2419 2588 2756 2924 3090 3256
  6 3420 3584 3746 3907 4067 4226 4384 4540 4695 4848
  7 5000 5150 5299 5446 5592 5736 5878 6018 6157 6293
  8 6428 6561 6691 6820 6947 7071 7193 7314 7431 7547
  9 : TABLE 50 0 DO 49 I - SINTABLE ! LOOP ; TABLE
 10  FORGET TABLE ( USE AND DISCARD )
 11 7660 7771 7880 7986 8090 8192 8290 8387 8480 8572
 12 8660 8746 8829 8910 8988 9063 9135 9205 9272 9336
 13 9397 9455 9511 9563 9613 9659 9703 9744 9781 9816
 14 9848 9877 9903 9925 9945 9962 9976 9986 9994 9998
 15 10000 -->
```

```
SCR # 16
  0 ( TRIG FUNCTIONS 2 )
  1 : TABLE 41 0 DO 90 I - SINTABLE ! LOOP ; TABLE
  2 FORGET TABLE ( USE AND DISCARD )
  3 : SIN180 DUP 90 > IF 180 SWAP - THEN SINTABLE @ ;
  4
  5 : SIN DUP DUP ABS / SWAP ABS
  6  360 /MOD DROP DUP 180 > IF 180 - -1 SWAP
  7    ELSE 1 SWAP  THEN SIN180 * * ;
  8
  9 : COS 90 + SIN ;
 10
 11
 12 ;S
 13
 14
 15
OK
```

MEMORY TESTER FOR CASIO FX-702P

This is a fairly simple Memory Test game in which the computer generates an increasingly large random number (displayed on the screen for an increasingly large period of time) which you, after a ten second pause, have to input back. The computer allows you three tries to input the correct number, but three 'strikes' and you're out.

Patrick Mead
Ashgrove QLD

```
 1 C=0
 5 VAC
 6 Z=1
10 WAIT 20:PRT "**
   MEMORY TESTER
   **":PRT ""
15 INP "ANY NUMBER
   =",S
20 S=INT (RAN#*10)
   /199*100
25 IF S=0 THEN 20
30 N=INT (S*Z)
35 C=C+5
40 WAIT C:PRT "REM
   EMBER";N
49 WAIT 60:PRT "
      "
60 IF W=3 THEN 200
80 INP "WHATS THE
   NUMBER",S
90 IF S*N;WAIT 20:
   PRT "WRONG":W=W
   +1:GOTO 60
110 WAIT 20:PRT "RI
   GHT!":R=R+1:W=0
   :Z=11*Z
120 WAIT 20:PRT R;"
    RIGHT SO FAR":
   GOTO 20
200 WAIT 30:PRT "FO
   RGET IT!START O
   VER":GOTO 1
```

SAMPLE RUN

```
** MEMORY TESTER **

ANY NUMBER=?
3
REMEMBER 4

WHATS THE NUMBER?
4
RIGHT!
 1 RIGHT SO FAR
REMEMBER 33

WHATS THE NUMBER?
33
RIGHT!
 2 RIGHT SO FAR
REMEMBER 304

WHATS THE NUMBER?
403
WRONG
WHATS THE NUMBER?
430
WRONG
WHATS THE NUMBER?
340
WRONG
FORGET IT!START OVER
** MEMORY TESTER **

ANY NUMBER=?
```

Computers in Schools

AT LAST! A PUBLICATION FOR TEACHERS, PARENTS, STUDENTS . . . AND ANYONE CONCERNED WITH THIS VERY IMPORTANT ISSUE.

ISSUES:

Does your child really need a home computer?
It's become rather difficult to see through all the media hype! We provide down-to-earth answers to major questions (Will your child by handicapped without a computer? Or even handicapped by one?)

The push into the schools market
An impartial look at the wave of advertising power currently breaking over the (supposedly) lucrative schools market. Who really stands to gain — the kids, or computer manufacturers?

VIEWPOINTS:

What the experts say
Recognised authorities in the fields of both education and computing give their opinions.

How the teachers feel
Computers are likely to feature strongly in the future of education. We tell how teaching staff are coming to grips with this fact.

What the kids think
Are computers worthwhile? Just good fun? Or a waste of time? Students comment (candidly) on how they feel about computer education.

PRACTICALITIES:

How to pick the right system
For high schools, primary schools.

Educational software
What's available; what's really educational; what to look for when shopping.

Networking and schools
What it is; how it works; how it can help.

Computer aided education in practice
Getting right down to basics: how are computers actually used in the schoolroom?

Computers in administration. The out-of-the-classroom aspect. How computers can streamline the behind-scenes running of a school.

CASE HISTORIES:

The UK scene
How far has computer education advanced overseas? And how far will it go?

True-life stories
From schools around Australia — how computers are working for them.

All this, plus a useful directory of services, and a simple-language glossary to help you in your purchase of a computer.

AVAILABLE JANUARY, 1985

COMPUTER CLUB LIST

ACT

ACT Micro 80 Users Group, Bill Cushing, 10 Urambi Village, Kambah, 2902, 062 313630.
ACT Vic 20 Users Association, Chris Groenhout, 25 Kerferd St, Watson, 2602, 062 41 2316, Meetings 1st Monday each month at Boy's Grammar Scout Hall, Red Hill, 7.30 onwards.
ACTARI, Chris McEwan, Co-Ordinator, ACTARI, P.O. Box E112, Canberra, 2600, 062 88 7861.
Apple User Group (ACT), Jeff Brock, 1 Buckley Circuit, KAMBAH, 2902, 062 313630.
Australian ZX80 Users Group(AZUG), David Brudenall, 19 Godfrey Street, Campbell, 2601, for ZX80/Microace owners.
Canberra ACT Sirius User Group, Jim Bland, 062 81 2824, 062 81 2832.
Canberra Compucolor Club (CCC), Meets 7.30 on first Sunday of every month at the offices of Digital Equipment, 28 Lonsdale Street, Braddon ACT.
Canberra Microbee Users Group, Hugh Gibson, Microbee Store, Level 1, Cooleman Court, Weston, 2611, 062 88 6384.
Canberra Microbee Users Group, Adrian Van Wierst, 9 McGowan Street, Dickson, 062 49 7030.
Canberra Micro-80 User Group, Milt Cottee, 33 Crawford Cres, Flynn, 2615, 062 58 8822, meetings third Monday each month 7.30 pm in the small theatrette, Reid TAFE, for System 80, TRS-80 etc.
Canberra NEC Users Group, Mal Smith, PO Box 173, Belconnen 2616, meets first Tuesday each month at Main Conference Room, CSIRO Headquarters, Limestone Avenue at 7.30, (062) 54 1614.
Canberra Osborne Group, c/o Geoff Cohen, P.O. Box 136, Kippax, 2615, 062 54 7608.
Micsig, Registrar, P.O. Box 446, Canberra, 2601.

NT

Alice Springs Microbee Users Group, Douglas Craigie, c/- PO Box 3230, Alice Springs 5750.
Darwin Microbee Users Group DBUG, Felino Molina, P.O. Box 3111, DARWIN, 5794, 089 82 5613bh, 089 88 1455ah.
N.T. Computer Club, Ian Diss, meets at Wulagi Primary School on the first and third Thursday of each month at 7.30. Users of all machines and other interested parties welcome, (089) 27 9208.
N.T. 80 Computer User Group, R T O'Brien, 433 McMillans Road, JINGILI, DARWIN, 5792.
The Microcomputer Assoc. of the N.T, Andy Smith, Darwin Community College, CASUARINA, 5792.
VZ-200 Users Club, 7 Abbott Crescent, Malak, Darwin 5793, (089) 272830.

SA

AACC, Adelaide Atari Computer Club, meets at Gilles Street Primary School, City, on first Monday (second if first is on Public Holiday) of each month. Secretary, PO Box 333, Norwood, SA 5067.
Adelaide Lotus 1-2-3 User Group, Paul Wragg, Pannell Kerr Foster, GPO Box 1969, Adelaide.
Adelaide Micro User Group, R. G. Stevenson, 36 Sturt Street, Adelaide, 5000, for TRS-80 and System 80 Users.
Adelaide Osborne Group, Russell Barter, The Secretary, 410 Regency Road, PROSPECT, 5082.
Beebnet, BBC and Econet User Group P.O. Box 262, KINGSWOOD, 5062, The group intends to produce a newsletter on a monthly basis. It is interested in any software producers or distributors who would be interested in serving the groups market requirements.
Commodore/Vic Computer Users Assoc., Mr Eddie Hann, 13 Miranda Road, PARALOWIE, 5108, The SA branch meets monthly.
Compucolor-Intecolor User of S.A., P.O. Box 86, Torrensville, 5031, 08 352 3296.
DEC Personal Computer Special Interest Group, see NSW entry.
IBM-PC S.A. Users' Group, PO Box 68, Walkerville 5081.
Kaypro User Group, Myles Wakeham, 100 Pirie Street, Adelaide, 5000, 08 223 6333, meetings 1st Tuesday each month.
Microbee Users Group of S.A. MUGSA, The Secretary, GPO Box 767, Adelaide 5001.
S.A. Commodore Computers U.G., Eddie Hann, The Secretary, P.O. Box 427, North Adelaide, 5006, 258 6367, meetings second Tuesday each month, 7.30 at Royal Caledonian Hall, 379 King William St, Adelaide.
S.A. Foundation for Computer Literacy, Michael Kennett, PO Box 210, Norwood 5067, caters for children from 6 years (unaccompanied) or 4 years with older friend or brother or sister. Special emphasis on the needs of handicapped, and educably disabled and socially disadvantaged children, but ALL children welcome. Family participation encouraged, phone (08) 51 5474.
S.A. Peach User Group, Geoff Drury, 27 Creslin Tce, Camden Park 5038, (08) 352 2555 or 295 2778 (ah), special interest group attached to the SA Microprocessor Group which holds separate meetings.
S.A. Microprocessor Group Inc SAMG, The Secretary, P.O. Box 113, Plymton, 5038, 08 278 7288.
Sorcerer Users Group of S.A., Don Ide, 14 Scott Road, Newton 5074.
South Australian Apple Users Club, The Secretary, SAAUC, C/- The Bookshelf, 169 Pirie Street, Adelaide, 5000.
South East Computer Enthusiasts' Group, Glenn Mibus, 3 Millard St, Mount Gambier 5290, 087 25 1046, meetings 2nd and 4th Tuesday of each month from 6.30 at Mt Gambier High School Computer Room, for all machines and interested parties.

COMPUTER CLUB LIST

NSW

Albury-Wodonga Dist Mbee U.G., Eric Eulenstein, 202 Kooba St, Albury, 2640, 060 25 1601.

Apple Users Disk Exchange Club, Peter Lapic, 45 Malabar Street, Canley Vale 2166.

Apple Users Group, Colin Rutherford, P.O. Box 505, Bankstown, 2200, meets 6.30 pm second Monday of each month (Tue after pub. hol.) at Sydney Grammar School, Stanley Street, Sydney, 02 520 0926.

Atari Computer Enthusiasts, Tony Reeve, PO Box 4514, Sydney 2001.

Ausborne, Brian Carney, 477 4492, P.O. Box C530 Clarence Street, Sydney, 2001, meetings third Wednesday each month at 6.30 in the North Shore Council Chambers, for Osborne users.

Ausbug, Stephen Ford, P.O. Box 62, Londonderry, 2753.

Australasia ZX80 Users Group, Tony Mowbray, 87 Murphys Ave, Kieraville, 2500, 042 28 5296, for ZX80/81 Microace owners.

Australasian ZX80 Users Newsletter, 87 Murphys Ave, Kieraville, 2500.

Blue Mountains Microbee Computer Club, Roger Cooper, 047 58 7238.

Blue Mountains Computer Club, Eric Lindsay or T. Macindoe, C/- P.O. Faulconbridge, 2776.

Broken Hill Microbee Users Group, Peter Cotter, 533 Radium Street, Broken Hill, 080 881621.

Central Coast Apple Users Group, C.W. Lee, 662 The Entrance Road, Wamberal 2260, meetings first Tuesday each month at the Niagara Park Public School from 7.30 pm, (043) 84 3419.

Central Coast Computer Club, Max Maughen, P.O. Box 36, Ettalong Beach, 2257, 043 24 2711, 1st and 3rd Tuesday every month at Applied Technology, West Gosford, for all types of computer.

Commodore Users Group, John Guidice, G.P.O Box 4721, Sydney, 2001.

Compucolor Users Group, Tony Lee, 52 Cowan Road, St. Ives 2075, phone (02) 449 8824.

Cumberland Computer User Group, S. O'Neil, 02 682 3851.

DEC Personal Computer Special Interest Group, Marion Rhydderch, DEC Australia, Northern Tower, Chatswood Plaza, Railway Street, Chatswood 2067, 02 412 5252.

Dubbo and District Microbee Users Group, Coralie Taylor, 18 Cunningham Street, Dubbo 2830, meets 4th Wednesday each month at 7.30 in the Dubbo High School Computer Room.

A.P.F. Users Group, Norm McMahon, 288 Kissing Point Road, TURRAMURRA, 2074, 02 44 2645.

Hawkesbury Commodore Computer Club, Richard Farrell, 12 Inverary Drive, Kurmond 2757, meets 4th Tuesday of each month at 7.30pm at Neighbourhood Centre, West Market Street, Richmond.

Hawkesbury MicroBee Computer Club, Bruce Rennie, 045 67 7329.

HP Desktop Computer Users Group, Dr. R. W. Harris, CSIRO Division of Mineral Physics, PMB 7, Sutherland 2232, 02 543 3460

Hunter U. G.- All Microcomputers, Secretary, P.O. Box 39, BROADMEADOW NSW, 2298, Meets on the second Wednesday of each month in Room 308, building W, University of Newcastle at 7.45pm. Membership is primarily Apple II orientated, but anyone with interest in micros welcome.

Illawarra Microbee Computer Club, Ronald Read, 49 Beatus Street, Unanderra, 2526.

Illawarra Super 80 Users Group, Jim O'Grady, Chairman, P.O. Box 1775, Wollongong, 2500.

Kaypro Users Group N.S.W., Harry Richards, 4/2 Bortfield Drive, Chiswick, 2046, 02 713 1585, meets 2nd Tuesday each month at 8.00 pm in the Burwood R.S.L.

Sydney Lotus 1-2-3 User Group, Ron Pollak, (02) 29 5316.

Macarthur Computer Association, J Napier, 23 Athel Tree Crescent, Bradbury 2560, meets first Monday each month at Airds High School, Briar Road Campbelltown at 7.30 each month, all machines are catered for, 046 25 2055.

Macquarie Microbee Users Group, Brian Thompson, meetings first Monday each month at Denistone East Primary School at 7.30 pm, 02 85 1659 after hours.

MEGS (Microcomputer Enthus. Group), John Whitlock, P.O. Box 1309, Chatswood 2067. Meetings third Monday each month at rear of St. Andrew's Presbyterian Church, 37 Anderson Street, Chatswood, (02) 638 1142.

Mi Computer Club, Norma Jackson, P.O Box 21, Waterloo, 2017, 02 662 8888.

Microbee Users Club (Broken Hill), Peter Cotter, 533 Radium Street, Broken Hill 2880, 080 88 1621.

Newcastle Microbee Users Group, Lee Osman, 12 Cleverton Close, Warners Bay 2282, 049 48 8813.

Newcastle Microcomputer Club, Angus Bliss, PO Box 293, Hamilton 2303, meetings 2nd and 4th Monday each month at room G12, Physics Building, Newcastle Uni, 049 67 2433.

N.S.W. Primary School Microbee Users Group, Mr Peter Stretton, c/- Hunters Hill Primary School Alexandra Street, Hunters Hill 2110.

N.S.W. 6800 Users Group, 27 Georgina Ave., Keiraville, 2500.

Northern Beaches Vic User Group, E. Tuxford, 161 Barrenjoey Rd., Newport, 2106, Ph 997 2467, Community Centre (If We're lucky).

Northern N.S.W MICC Chapter, Alen Hartley, Dundurrabin via Dorrigo, 2433, 066 57 8160.

N.S.W Peach User Club, Daniel Soussi, 02 698 8286, weekly meetings on Saturday from 2pm at 'Cybernetics Research' 120-122 Lawson St Redfern.

OSI Users Group, Nigel Bisset, 02 411 7142.

Pocket Computer Users Club, George Antonijevic, 02 683 4296, for those interested in pocket computers, whatever the brand. Meetings held on the first Wednesday of each month at 7.30pm at the 'Woodstock' Community Centre, Church St. Burwood.

Sorcerer Users Group, P.O. Box E162, St James, 2000, meetings 1st Tuesday each month at 7th Floor Datec House, 220 George Street, Sydney at 7.30pm.

Southern Districts Commodore Users Group, Lex Toms, 602 8691, 3 Lucille Crescent, Casula 2170, Meetings 1st and 3rd Wednesday each month, API Hall Currajong Road, Prestons.

Sutherland Super 80 Group, Jim Traeger, 02 525 2018, Super 80.

Sydcom 64 (C64 User Group), Andrew Farrell, meetings first Tuesday of each month at 6.30 pm above Computerwave, George Street, Sydney, 02 99 2640.

Sydney Forth Group, Peter Tregeagle, 10 Binda Road, Yowie Bay, 2228, 02 524 7490, meets 2nd Friday of each month at 7.00pm in the John Goodsell Building, UNSW room LG19.

Sydney MicroBee Users Club, Colin Tringham, 92 6408, PO C233, Clarence St, Sydney 2000, Meetings 3rd Sat each month 1-5 pm McMahons Point Hall, Blues Point Rd North Sydney.

Sydney Peach User Group, Ben Sharif, 261 Northumberland Street, Liverpool, 2170, 02 601 8493.

Sydney TRS-80 Users Group, meetings 2nd, 3rd and 4th Saturday of each month at Botany, phone (02) 666 4716 bus hours.

TAG-The Access Group, Bob Dolton, PO Box 943, Orange 2800, for Access and Actrix users.

T.I. Sydney Home Computer U.G., P.O. Box 149, Pennant Hills, 2120.

Wagga Microbee Users Group, John Simmons, 47 Undurra Drive, Glenfield 2650, 069 31 1302, meetings 1st and 3rd Tuesdays each month in the Tolland-Glenfield Neighbourhood Centre at 8.00pm.

Wizzard User Group, John Mifsod, 150 Bouganville Road, Blackett, 2770, 02 628 0801.

ZX-Spectrum Users Club, Craig Kennedy, P.O. Box 466, Epping, 2121.

QLD

Adventure Club, Christine Ogden, 37 Samford Road, Leichhardt, Ipswich 4305, for all Adventure type game players.

Apple-Q the Brisbane User Group, The Secretary, P.O. Box 721, SOUTH BRISBANE, 4101, Has User Group days every third Sunday of month at Hooper Education Centre, Kuran St. Wavell Heights. Centre is open from 8.30am till 4.30pm, members encouraged to bring Apple along.

Australian Sirius Users Group, P.O. Box 204, CHERMSIDE, 4032, 07 350 2611, Looks after the needs of Sirius One and Victor 9000 computer users. For membership form write to above address.

Basic User Group, Chris Lucey, Cranium Computers, 34 Lawless Street, Blackwater 4717.

Brisbane Medfly Users Group, K.J. Walker, 120 Highgate Street, Coopers Plains 4108.

Brisbane Sinclair (Spectrum) Computer Club, V. Lewis, 37 Samford Road, Leichhardt Ipswich 4305, meets third Sunday at Everton Park State High School, at 2.00, 07 355 7809.

Brisbane Super 80 Users Group, Gary Gatfield, 08 355 3173.

Brisbane Youth Computer Group, A. Harrison, P.O. Box 396, Sunnybank, 4109.

Cairns District Microbee Users Group, Chas Eustance, 21 Marr Street, Edmonton 4869, (070) 554531.

Commodore Computer Users Group QLD, Mrs D D Dillan, P.O. Box 127, STONES CORNER, 4120.

Commodore Users Group, John Egan, P.O. Box 274, SPRINGWOOD, 4127, 07 287 2705, Is for owners of Pet/CBM and Vic-20 machines. Meetings held on the first Tuesday of the month at 130 Petrie Terrace, Brisbane.

Computer Owner's Group, Betty Adcock, 42 Lucan Ave, Aspley, 4034, 263 4268, 2nd Wednesday each month, 7.45 pm, all kinds of computer are catered for.

DEC Personal Computer Special Interest Group, see NSW entry.

Gold Coast Microbee User Group, Col McLaren, 1-100 Imperial Parade, Labrador, 4215, 075 314610, meetings first Sunday each month, 3.00 at the Southport High School.

IREE Microcomputer Interest Group, N Wilson, P.O. Box 811, ALBION, 4010.

Mackay Microbee User Group, Geoff Gehring, Box 230, Mackay, 4740, 079 42 3214.

Osborne Users Group of Qld Uni, Glen McBride, meetings 2nd Thursday each month open to all, 07 371 4243.

Superboard Users Group, Ed Richardson, 146 York Street, NUNDAH, 4012.

Tandy, Apple, Commodore UG, Chris Lucey, 34 Lawless Street, Blackwater 4717.

The Microcomputer Society, The Secretary, P.O. Box 580, FORTITUDE VALLEY, 4006, Meetings are held on the second Friday of each month in the Old Town Hall, corner Vulture and Graham Streets, Sth Brisbane. Meetings start at 7.30pm if main gate is closed use the back stairway.

Townsville MicroBee User Group TMUG, Mannie Van Rijswijk, PO Box 5751 M.C., Townsville 4810, meetings 7.30 pm on second and fourth Monday each month on the Ground Floor, St Margaret Mary's Secondary School, Crowle Street, Hermit Park.

TRS80/System 80 Computer Group, Secretary, 16 Laver Street, Macgregor 4109, (07) 343 5771, meets first Sunday each month at Lindum Hall, Lindum Street, Lindum at 2.00pm.

ZX 81 Club, P. Carswell, 22 Braud Street, BUNDABERG, 4670.

NZ

1802 Users Group, P.O. Box 6210, AUCKLAND, NEW ZEALAND, For those who own an ETI-660 or a COSMAC VIP, you can contact the 1802 Users Group. Be kind and send them a return addressed envelope and some International Reply Coupon.

Nelson Vic Users Group, Peter Archer, Nelson VIC Users Group, C/o P.O. Box 860, Nelson N.Z., for Vic and Commodore.

Wellington Microcomputer Soc. Inc, Lindsay Williams, 2 Pope Street, PIMMERTON, NEW ZEALAND.

ZX81 Club, R Skelton, C/- Harbourside Orchard, WAIUKU NEW ZEALAND.

TAS

***DEC Personal Computer Special Interest Group,** see NSW entry.

Devonport Computer Interest Group, John Steveson, R.S.D 422, SHEFFIELD TASMANIA, 7306, 004 92 3237.

Spectravideo Computer Users Group, Mr W. P. Decket, 48 Heather Street, LAUNCESTON, 7250, 44 4836, Membership to the club costs $15 which entitles members to a newsletter and to discounts on computer equipment.

Tasbeeb, John Hannon, PO Box 25, North Hobart 7000, meetings first Monday each month at Elizabethan Matriculation College in D Block at 8pm, 002 34 2704, for BBC computers.

Tasmanian T.I. User Group, Coordinator, 1 Benboyd Court, ROKEBY, 7019, 002 29 4009, meetings third Sunday of each month at University of Tasmania, room 373.

TAS-Micro, Peter Deckert, Unit 1/456 West Tamar Road, RIVERSIDE, LAUNCESTON, 7250.

Tasmanian Commodore Users Assoc., Vincent T. Staggard, The Secretary, G.P.O. Box 391D, Hobart, 7000, 002 72 0295, Commodore and others.

Tasmanian OSI User Group, David Tasker, 111 Bass Highway, WESTBURY, 7303.

COMPUTER CLUB LIST

VIC

Apple Users Society of Melbourne, D. Halprin, 03 387 3221, PO Box 43, Forest Hill 3131.

AT Microcomputer Club, Grant Forest, 03 8792257ah, 03 699 2888 bh. This club has been formed for people interested in the Applied Technology DGOS Z80.

Atari User Groups Melbourne, Kelvin Eldridge, P.O. Box 173, 3073.

Australian Forth Interest Group, Tony Latermore, P.O. Box 704, SALE, 3850, 051 44 2011.

Australian North Star Users Assoc., P.O. Box 194, WANGARATTA, 3677.

Ballarat Computer Users Group, Publicity Officer: John Preston, 053 31 4363.

Billanook Computer Forum, Mr Maurie Canterbury, Cardigan Road, Mooroolbark 3138, (03) 725 5388.

BUG 80 (Burwood Users Group), P.O. Box 46, BLACKBURN SOUTH, 3130.

Chip 8, 6800, 1802 User Group, Frank Rees, 27 King Street, BOORT, 3537.

Compucolor Users Group, L Ferguson, 12 Morphett Avenue, ASCOT, 3342.

DEC Personal Computer Special Interest Group, see NSW entry.

Forth Interest Group, Lance Collins, P.O. Box 103, CAMBERWELL, 3124, (03) 29 2600, Meets on the first Friday of the month at the Bowen Street Neighbourhood Centre, 102 Bowen Street, Camberwell South.

Geelong Commodore Computer Club, D Gerrard, 15 Jacaranda Place, Belmont 3216, (03) 44 2863.

Geelong Computer Club, Peter McKeon, P.O. Box 93, GEELONG, 3220.

IBM & Columbia Computer Users Club, Giles Bray, 22/11 Auburn Grove, Hawthorn East, 3123, 82 7632, 2nd Tuesday each month, 7.30 at the Victorian College of Pharmacy.

Kaypro Users Group of Victoria, George Kunz, PO Box 159, Forest Hill 3131, 03 857 5462, meetings fourth Sunday each month at Burwood State College Community Resources Centre at 2 pm.

KAOS (Ohio Scientific), David Anear, 49 Millewa Crescent, DALLAS, 3047.

Latrobe Valley Colour Computer U.G., George Francis, 31 Donald Street, Morwell, 3840, 22 1389, for TRS-80 & MC10 users.

Melbourne Atari Computer Enthusiast, PO Box 133, Mulgrave North 3170, meetings held on first Sunday of each month at 11.40am at Monash University Rotunda.

Melbourne Lotus 1-2-3 Users Group, Robert Taylor, (03) 267 4800.

Melbourne MicroBee Users Group, Pres Grant Forrest, PO Box 157, Nunawading 3131, meetings 7.00 pm second Wednesday each month at VIC State College-Burwood Campus, 221 Burwood Highway, Burwood.

Melbourne PC User Group, Stephen Wagen or Christopher Leptos, c/o Pannell Kerr Foster, 14th Floor, 500 Bourke Street, Melbourne 3000, phone (BH) (03) 605 2222.

Melbourne Peach Users Group (MPUG), P.O. Box 191, Rosanna, 3084, 03 434 2541.

Melbourne Super 80 Users Group, Hon. Sec. Victor Shuttleworth, 03 723 2713.

MICOM, Microcomputer Club of Melb., P.O. Box 60, CANTERBURY, 3126.

National Mutual Micro Users Group, R Prewett, NMLA, PO Box 2830AA, GPO Melbourne 3001, for National Mutual staff.

National Sinclair User Group, P.O. Box 148, GLEN WAVERLEY, 3150.

National ZX80 Users Club, 24 Peel Street, COLLINGWOOD, 3066.

NEC Portable User's Group, D Green, meetings second Wednesday of each month at Myers Computer Centre Lonsdale Street at 7.30 pm, (03) 611 3380.

Northn/Westn Sub. Comp. Users Group, John King (Secretary), 284 Union Road, MOONEE PONDS, 3039, 03 338 9304, Contact CP/M Data Systems.

Peninsula Computer Club, George Thompson, 3 Patterson Street, Bonbeach, 3196, 772 2674, 2nd Tuesday each month at Chisholm College, Frankston, many types of computers are catered for.

Sharp Computer Users Association, The President, 7 Faye Street, East Burwood 3151.

Spectravideo Users Group, Mitch Raitt, Fernhill, Tindal's Road, Warrandyte 3113, (03) 844 3485.

Sorcerer Computer Users (Australia), Secretary, G.P.O. Box 2402, MELBOURNE, 3001.

TI-99/4A Users Group Melbourne, Wayne Worladge, 123 Ashburn Grove, Ashburton, 03 25 1832.

The Motorola User Group Soc. (MUGS), Clive Allan, 11 Haros Avenue, NUNAWADING, 3131, 03 878 1298, Group is interested in 6800/02/09 based computers, particularly if running Flex although this is not a prerequisite to join.

Vic. Assoc. of Computer Educators, Arthur Totrall, P.O. Box 69, WHITTLESEA, 3757.

Victorian VZ200 User Group, Luigi Chiodo, 24 Don St., Reservoir, 3073, 03 460 3770.

Victorian Wizzard Users Group, Barry Klein, 24 Russell Street, Bulleen 3105.

Yarrawonga Computer User Group, Chris Younger, 057 44 3859, 10 Witt Street, Yarrawonga, 3730, for all machines.

ZX81 Software Exchange, C/- Chips Taens, 5 Muir Street, MT. WAVERLEY, 3149.

WA

Agriculture Users Group, c/- Mr R Fenwick, Dept. of Agriculture, Albany 6330. For farmers and the agriculture service industries.

CU WEST WA Compucolor/Intecolor U.G, John Newman, 8 Hillcrest Drive, DARLINGTON, 6070.

DEC Personal Computer Special Interest Group, see NSW entry.

KAOS-W.A.,Gerry Ligtermoet, 09 450 5081, 39 Cloister Ave, MANNING, 6152, for Ohio Scientific Users.

OSWEST-Osborne Users Group of W.A., Mal Ferguson, PO Box 199, Mundaring 6554, meets first and third Wednesday at the Palmyra Recreation Centre and the Subiaco Exhibition Hall respectively from 7.30, 09 295 1449, for Osborne and other interested computer users.

Kaypro User Group of WA, Ainslie Sharpe, PO Box 91, Claremont 6010, 09 384 5511, meetings 2nd and 4th Mondays of each month in the Canteen of the Department of Agriculture, Jarrah Road, South Perth.

Perth 80 Users Group, C Powell, 09 457 6849, for System 80 and TRS 80 Users.

Perth Hitachi Peach Club, The Secretary, 1 Charf Court, Riverton, 6155, 09 367 5880, for Hitachi Peach & 6809s.

Sorcerer Computer Users of Aust., The Secretary, 90 King George Street, PERTH SOUTH, 6151, 09 367 6351.

Super 80 Users Group Perth, Garry Black, 19 Bendigo Way, CITY BEACH, 6015, 09 385 8813.

The W. A. Atari Computer Club, Mr Alf Gaebier (Secretary), P.O. Box 7169, Cloisters Square, PERTH, 6000.

W.A. Microbee Club, Mike Oborn, 09 447 5366.

Vic-Ups, G. Padfield, 09 451 4629.

W.A. Wizzard Users Group, John REid, 13 Wenlock Road, Wattleup 6166, 09 410 2359.

W.A. ZX Users Group, Phil Taylor, 09 328 4111, (bh).

WA University Computer Club, 2nd Floor, University of WA, Guild Building, 09 386 1455.

UNLEASH THE MUSIC IN YOUR COMMODORE 64

SIGHT & SOUND
Music Software, Inc.

THE INCREDIBLE MUSICAL KEYBOARD

Now with no musical experience you can unleash the music that's hidden deep within your computer. This unique keyboard comes complete with its own software that will get you playing right away. It even includes two music books, one for the beginner, and one for the more advanced musician.

KAWASAKI SYNTHESISER

This transforms your computer into a programmable synthesiser and sound processor. Use the polyphonic keyboard and preset sounds to create your own music. Unique three-track recording capabilities to produce an unlimited number of instrument sounds and special effects.

KAWASAKI RHYTHM ROCKER

Create space sounds and hi-resolution colour graphics to the beat of electronic rhythms. This creative Music/Video synthesiser offers a variety of percussion and synthesiser sounds that can be laid down on top of pre-programmed bass lines.

MUSIC PROCESSOR

Create, edit, record, play and compose your own musical arrangements. Use your joystick to change any of the 99 preset instrument and special effect sounds. Full music editor allows you to create new music or change one of the 16 pre-recorded tunes. See visual musical notation on the screen. Music Processor is compatible with the entire computer song album series.

TUNE TRIVIA

Challenge yourself and your friends to music trivia from an almost infinite number of songs and questions. Four fascinating games in one, add your own tunes and trivia questions, try to stump your friends. Tune Trivia comes complete with superstar songbook and full documentation.

COMPUTER SONG ALBUMS

Listen to the hits of Michael Jackson, Willie Nelson, Denise Williams, The Eurythmics, while you control computer generated instruments and special effects. Every musical arrangement comes complete with bass, rhythm, melody and harmony. Computer Song Albums are also compatible with the music video kit soon to be released.

Ozi Soft™

AUSTRALIAN DISTRIBUTORS.
(02) 211-1266

FUTURE 1984 RELEASES INCLUDE:

Music Video Kit –
Make your own music videos with this powerful graphics processor. Paint a background scene, then syncronize your masterpiece to your music.

3001 Sound Odyssey –
Travel through a complete tutorial on musical synthesis. Learn how subtle change in A.D.S.R., waveforms, ring modulation and filters work together, then create your own music.

AVAILABLE AT ALL GOOD COMPUTER STORES ...